ACOUSTIC DATA PROCESSING USING THE DECENTRALIZED SQUARE ROOT INFORMATION FILTER

Final Report

for

September 01, 1988 through February 28, 1989

Sponsored by

Defense Advanced Research Projects Agency
Defense Small Business Innovation Research Program
ARPA Order No. 5916, Amdt 9
Issued by U.S. Army Missile Command Under
Contract #DAAH01-88-C-0412

Prepared by

M.R. Belzer, Y.M. Cho

Mentor Technologies, Inc.
318 Wall Street, Suite 2B
Kingston, New York 12401
(914)338-1680

February 28, 1989

"The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either express or implied, of the Defense Advanced Research Projects Agency or the U.S. Government."

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1a. REPORT SECURITY CLASSIFICATION	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: "Approved for public release; distribution unlimited"
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION: Mentor Technologies, Inc. / 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: Defense Advanced Research Projects Agency
6c. ADDRESS (City, State, and ZIP Code): 53-50 206th Street, Bayside, New York 11364	7b. ADDRESS (City, State, and ZIP Code): Commander, U.S. Army Missile Command, ATTN: AMSMI-RD-DP-T, Redstone, Arsenal, AL 35898-5244
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: DARPA / 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: DAAH01-88-C-0412
8c. ADDRESS (City, State, and ZIP Code): 1400 Wilson Boulevard, Arlington, VA 22209-2306	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification)
Acoustic Data Processing Using the Decentralized Square Root Information Filter

12. PERSONAL AUTHOR(S)
M.R. Belzer, Y. M. Cho

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final Technical	FROM 01Sep88 TO 28Feb89	89 February 28	68

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	undersea surveillance, distributed filtering, integrated tracking

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

see next page

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL: Mr. Hugh Carson	22b. TELEPHONE (Include Area Code): (205)876-3406 / 22c. OFFICE SYMBOL: AMSMI-RD-DP-T

DD Form 1473, JUN 86 — Previous editions are obsolete — SECURITY CLASSIFICATION OF THIS PAGE

ACOUSTIC DATA PROCESSING USING THE DECENTRALIZED SQUARE ROOT INFORMATION FILTER

Project Summary

Very Large Scale Integration (VLSI) technology has been developed to the point where special purpose processors may be concatenated to form supercomputers with far greater throughput rates than uniprocessor machines. MTI has developed a parallel form of the conventional Kalman filter that is well suited to being implemented in a multiprocessing environment. Moreover, our Decentralized Square Root Information Filter (DSRIF) has several very unique features which could be incorporated into the design of an integrated undersea tracking system with much improved performance over existing methods.

Phase I research demonstrated feasibility of the DSRIF as a means for solving the linear least squares estimation problem in decentralized form. Underwater tracking of a high velocity torpedo (undergoing high dynamic maneuvers) was simulated. Also, an extended form of the DSRIF was used to complete the processing of real Multiple Rocket Launch System Data provided by the White Sands Missile Range. In this case, an adaptive form of the extended DSRIF, wherein the process noise levels were made to be a function of the globally optimal estimate error covariance, was successfully used to track the rocket data.

TABLE OF CONTENTS

LIST OF SYMBOLS

i	superscripted local system number
j	superscripted vector element number
k	time index, may be subscripted or enclosed in parentheses
x_k	global state vector
w_k	global process noise vector
Φ_k	global state transition matrix
x_k^i	local state vector
${}_o x^i$	origin of a local coordinate system in earth centered earth fixed coordinates
${}_o x$	origin of the global coordinate system in earth centered earth fixed coordinates
${}_o r^i$	radial vector from earth's center to the origin of a local coordinate system
w_k^i	local process noise vector
Φ_k^i	local state transition matrix
y_k^i	local measurement vector
C_k^i	global observation sub-matrix
H_k^i	local observation matrix
v_k^i	local measurement noise vector
$x_k(\pm)$	$\left(\begin{array}{l}\text{measurement updated}\\ \text{time updated}\end{array}\right)$ global state estimate
$x_k^i(\pm)$	$\left(\begin{array}{l}\text{measurement updated}\\ \text{time updated}\end{array}\right)$ local state estimate
$P_k(\pm)$	$\left(\begin{array}{l}\text{measurement updated}\\ \text{time updated}\end{array}\right)$ global estimate error covariance matrix
$P_k^i(\pm)$	$\left(\begin{array}{l}\text{measurement updated}\\ \text{time updated}\end{array}\right)$ local estimate error covariance matrix
M_k^i	matrix of zeros and ones which partition the global states to the local systems
Q_k	covariance matrix for process noise vector

R_k^i covariance matrix for local measurement noise vector

$P_0(-)$ initial global estimate error covariance matrix

N number of time samples

M number of local systems

M^i number of local measurement variables

$z_w(k)$ "pseudomeasurement" vector

$R_w(k)$ inverse square root of Q_k

$R_k(\pm)$ (measurement updated / time updated) global square root information matrix

$z_k(\pm)$ (measurement updated / time updated) global square root information vector

$R_k^i(\pm)$ (measurement updated / time updated) local square root information matrix

$z_k^i(\pm)$ (measurement updated / time updated) local square root information vector

e_k^i local "innovations" vector

$R_w^{*(i)}(k)$ $R_{wx}^{*(i)}(k)$ $z_w^{*(i)}(k)$ local smoothing coefficients

$R_w^*(k)$ $R_{wx}^*(k)$ $z_w^*(k)$ global smoothing coefficients

$z^*(k)$ $H^*(k)$ merge coefficients

t continuous time variable

d^i local coordinate system displacement vector

L latitude of global coordinate system origin

δ longitude of global coordinate system origin

α^i, β^i, τ^i Euler angles defining the orientation of the local coordinate system w.r.t. to the global coordinate system

r_e average earth radius

$^1T_k^i$ orthogonal transformation in local time update

$^2T_k^i$ orthogonal transformation in local measurement update

^3T_k orthogonal transformation in first merge step

^4T_k orthogonal transformation in second merge step

T^i orthogonal coordinate transformation between the local and global coordinate system

T^g orthogonal coordinate transformation between the global coordinate system and the earth centered earth fixed coordinate system

μ product of G and re

r_k^i range measurement

$\dot{r}_k^i$ range rate measurement

θ_k^i elevation angle measurement

Γ_k^i azimuth angle measurement

G gravitational constant

m_e earth's mass

Ω earth's angular velocity

u(t) deterministic control

f(x(t),u(t)) nonlinear dynamics vector

$h^i(x(t))$ nonlinear local observation vector

F_k linearized dynamics matrix

LIST OF FIGURES

Figure A: Stnd Dev of the Global Position Estimate Error Versus Time.
$R_k(j,j)=10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{17} 7.4×10^{17} 9.3×10^{17}] down to
$Q_k(j,j)$ = diag [1.7×10^{-18} 7.4×10^{-18} 9.3×10^{-18}]

Figure B: Stnd Dev of the Global Velocity Estimate Error Versus Time.
$R_k(j,j)=10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{17} 7.4×10^{17} 9.3×10^{17}] down to
$Q_k(j,j)$ = diag [1.7×10^{-18} 7.4×10^{-18} 9.3×10^{-18}]

Figure C: Stnd Dev of the Global Velocity Estimate Error Versus Time.
$R_k(j,j)=10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{17} 7.4×10^{17} 9.3×10^{17}] down to
$Q_k(j,j)$ = diag [1.7×10^{-18} 7.4×10^{-18} 9.3×10^{-18}]

Figure D: Stnd Dev of the Global Position Estimate Error Versus Time.
$R_k(j,j)=10^{10}$ down to 10^{1} for ot variables, and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{1} 7.4×10^{1} 9.3×10^{1}]

Figure E: Stnd Dev of the Global Velocity Estimate Error Versus Time.
$R_k(j,j)=10^{10}$ down to 10^{1} for ot variables, and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{1} 7.4×10^{1} 9.3×10^{1}]

Figure F: Stnd Dev of the Global Velocity Estimate Error Versus Time.
$R_k(j,j)=10^{10}$ down to 10^{1} for ot variables, and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{1} 7.4×10^{1} 9.3×10^{1}]

Figure G: Stnd Dev of the Global Position Estimate Error Versus Time.
$R_k(j,j)=10^{4}$ down to 10^{-1} for ot variables, and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{-3} 7.4×10^{-3} 9.3×10^{-3}]

Figure H: Stnd Dev of the Global Velocity Estimate Error Versus Time.
$R_k(j,j)=10^{4}$ down to 10^{-1} for ot variables, and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{-3} 7.4×10^{-3} 9.3×10^{-3}]

Figure I: Stnd Dev of the Global Velocity Estimate Error Versus Time.
$R_k(j,j)=10^{4}$ down to 10^{-1} for ot variables, and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{-3} 7.4×10^{-3} 9.3×10^{-3}]

Figure J: Stnd Dev of the Global Position Estimate Error Versus Time.
$R_k(j,j)=10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively. Q_k = factor * $P_k(+)$, where factor = 1.0 down to 10^{-3}.

Figure K: Stnd Dev of the Global Velocity Estimate Error Versus Time. $R_k(j,j)=10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively. Q_k = factor * $P_k(+)$, where factor = 1.0 down to 10^{-3}.

Figure L: Stnd Dev of the Global Velocity Estimate Error Versus Time. $R_k(j,j)=10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively. Q_k = factor * $P_k(+)$, where factor = 1.0 down to 10^{-3}.

Figure M: Stnd Dev of the Global Position Estimate Error Versus Time. $R_k(j,j)=10^{10}$ down to 10^{-1} for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively. Q_k = factor * $P_k(+)$, where factor = 10^{-3}.

Figure N: Stnd Dev of the Global Velocity Estimate Error Versus Time. $R_k(j,j)=10^{10}$ down to 10^{-1} for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively. Q_k = factor * $P_k(+)$, where factor = 10^{-3}.

Figure O: Stnd Dev of the Global Velocity Estimate Error Versus Time. $R_k(j,j)=10^{10}$ down to 10^{-1} for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively. Q_k = factor * $P_k(+)$, where factor = 10^{-3}.

LIST OF TABLES

1.0 Introduction and Problem Statement

The trajectory estimation problem is a problem of nonlinear estimation. Assume that the state of a target evolves in time according to the equation

$$\dot{x}(t) = f(x(t),u(t)) + w(t) \tag{1}$$

where u(t) is the target's nominal control vector and w(t) is a zero mean white noise process with spectral density Q(t). Values for the latter are selected in order to compensate for errors in the model which may originate from unknown perturbations to the nominal control (such as turbulence, strong currents, large surface waves, etc.) as well as from uncertainties in the hydrodynamics of the target vehicle. The corresponding discrete measurement vector for the i^{th} sensor is given by

$$y_k^i = h^i(x_k) + v_k^i \tag{2}$$

where v_k^i is a zero mean white noise sequence with covariance R_k^i. The problem is to estimate the target states x_k based upon all of the past measurements y_l^i where ($1 \le i \le M$ and $1 \le l \le k$). The state vector x_k contains the target position and velocity, biases which account for the displacement and orientation of the sensor or "local" coordinate systems (LCSs) w.r.t. the global coordinate system (GCS), and acceleration when the target vehicle is maneuvering.

Unfortunately, the optimal nonlinear estimator (conditional mean) cannot be realized with a finite-dimensional implementation and consequently, all practical nonlinear filters must be suboptimal. The usual suboptimal solution is the Conventional Kalman Filter when the nominal trajectory is known a priori, the Extended Kalman Filter when the nominal trajectory is unavailable, and Higher Order Filters (such as the second order filter [3], the single-stage iterative filter [4], and the Gaussian sum filter [5], among others [6]) when even greater accuracy is desired. The tradeoff here is performance versus real-time computational requirement.

Thus, we see that the ability of a sensor group to accurately record the motion of one or more underwater targets is a function of the individual target and sensor dynamical models as well as the particular algorithm used to combine raw data and produce track estimates. Another issue is intersensor communication. Sensors which operate independently from one another will exhibit larger estimate errors than ones which communicate with other members of the network.

Let the global discrete time linear system

$$x_k = \Phi_{k-1}\, x_{k-1} + w_{k-1} \tag{3}$$

where

$$w_k = N(0,Q_k) \tag{4}$$

$$x_0 = N(0,P_0(-)) \tag{5}$$

be the target dynamics linearized about the nominal trajectory (or estimate), and the global measurement model

$$\begin{bmatrix} y_k^1 \\ \cdot \\ \cdot \\ \cdot \\ y_k^M \end{bmatrix} = \begin{bmatrix} C_k^1 \\ \cdot \\ \cdot \\ \cdot \\ C_k^M \end{bmatrix} x_k + \begin{bmatrix} v_k^1 \\ \cdot \\ \cdot \\ \cdot \\ v_k^M \end{bmatrix} \tag{6}$$

where v_k^1 through v_k^M are uncorrelated random vectors and

$$v_k^i = N(0, R_k^i) \tag{7}$$

be the model for the tracking sensors.* The problem is to calculate the globally optimal (minimum mean square error) estimate of x_k and its associated estimate error covariance matrix P_k when y_k^1 through y_k^M are processed separately by locally optimal estimators 1 through M correspondingly. The local dynamical models are

$$x_k^i = \Phi_{k-1}^i \, x_{k-1}^i + w_{k-1}^i \tag{8}$$

where

$$w_k^i = N(0, Q_k^i) \tag{9}$$

$$x_0^i = N(0, P_0^i(-)) \tag{10}$$

and the local measurement models are

$$y_k^i = H_k^i \, x_k^i + v_k^i \tag{11}$$

where v_k^i satisfies (7). Notice that the local states may be physically different from the global states. Wilsky et.al. [7] have recently shown that a necessary and sufficient condition for our being able to recover globally optimal state estimates from locally optimal ones is that

$$C_k^i = H_k^i \, M_k^i \tag{12}$$

and for the tracking application, we expect that

$$x_k^i = M_k^i \, x_k \tag{13}$$

where M_k^i is a matrix which results in the correct partitioning of global states to the subsystem filters. Alternatively, equation (12) allows us to define the local state vector in terms of the local coordinate system. In this case, M_k^i is the orthogonal matrix which defines the transformation between local and global coordinate systems.

---

*Lower case variables are vector quantities while upper case generally corresponds to matrices of appropriate dimension. Also w_k, x_0, and v_k^1 through v_k^M are uncorrelated with each other for all k and $N(\alpha,\Gamma)$ signifies an α mean white Gaussian process with covariance matrix Γ.

1.1 The Decentralized Square Root Information Filter as a Solution

Decentralized processing is achieved by distributing the minimization of the conventional least squares performance criterion amongst the local filters and global merging equations that follow (see [1] or our Phase I proposal to DARPA for details). Let

$$M_k^i = I \tag{14}$$

The best distribution for multisensor tracking is probably to minimize $|| H_k^i xk - z_k^i ||^2$ in each of the M local filters and minimize the remaining two terms in the performance criterion within the central (merge) processor. However, this point will be explored in detail in Phase II research. Thus, each of the tracking sensors may be processed using the standard SRIF mechanization. For the ith sensor, we have

Measurement Update

$$ {}^1T_k^i \begin{bmatrix} R_k^i(-) & z_k^i(-) \\ (R_k^i)^{-\frac{1}{2}} H_k^i & (R_k^i)^{-\frac{1}{2}} y_k^i \end{bmatrix} = \begin{bmatrix} R_k^i(+) & z_k^i(+) \\ 0 & e_k^i \end{bmatrix} \tag{15}$$

Time Update

$$ {}^2T_k^i \begin{bmatrix} R_w(k) & 0 & z_w(k) \\ -R_k^i(+)\,\Phi_k^{-1} & R_k^i(+)\,\Phi_k^{-1} & z_k^i(+) \end{bmatrix} $$

$$ = \begin{bmatrix} R_w^{*(i)}(k) & R_{wx}^{*(i)}(k) & z_w^{*(i)}(k) \\ 0 & R_{k+1}^{(i)}(-) & z_{k+1}^{(i)}(-) \end{bmatrix} \tag{16}$$

where

$$[\, R_0^i(-) \quad z_0^i(-) \,] = [\, 0 \quad 0 \,], \tag{17}$$

$$[\, R_w(k) \quad z_w(k) \,] = [\, 0 \quad 0 \,], \tag{18}$$

the a priori estimate $\{ R_0^{-1}(-)\, z_0(-) \}$ has covariance

$$P_0(-) = R_0^{-1}(-)\, R_0^{-T}(-), \tag{19}$$

$$Q_k = R_w^{-1}(k)\, R_w^{-T}(k)\,, \tag{20}$$

$z_w(k) = R_w(k)$ times the a priori expectation of w_k, and the pair $\{ z_k^i, H_k^i \}$ correspond to normalized measurement equations i.e., $R_k^i = I$. Also, ${}^1T_k^i$ and ${}^2T_k^i$ are orthogonal transformations which put the matrices on the left hand sides of (15) and (16) respectively into upper triangular form. They may be implicitly computed using Householder transformations.

The local smoothing coefficients are combined with process noise and prior on x_0 by solving the following recursive equation:

$$ {}^3T_k \begin{bmatrix} R_w^{*(1)}(k) & R_{wx}^{*(1)}(k) & z_w^{*(1)}(k) \\ \vdots & \vdots & \vdots \\ R_w^{*(M)}(k) & R_{wx}^{*(M)}(k) & z_w^{*(M)}(k) \\ R_w(k) & 0 & z_w(k) \\ -H^*(k-1)\,\Phi_k^{-1} & H^*(k-1)\,\Phi_k^{-1} & z^*(k-1) \end{bmatrix} $$

$$ = \begin{bmatrix} R_w^*(k) & R_{wx}^*(k) & z_w^*(k) \\ 0 & H^*(k) & z^*(k) \\ 0 & 0 & \# \end{bmatrix} \tag{21} $$

$$ [\, H^*(-1) \;\; z^*(-1) \,] = [\, R_0(-) \;\; z_0(-) \,] \tag{22} $$

and $H^*(k)$ is upper triangular. To obtain the globally optimal information vector $z_k(+)$ and square root information matrix $R_k(+)$, we solve the following equations using $H^*(k)$ and $z^*(k)$ from (21):

$$ {}^4T_k \begin{bmatrix} R_k^{(1)}(+) & z_k^{(1)}(+) \\ \vdots & \vdots \\ R_k^{(M)}(+) & z_k^{(M)}(+) \\ H^*(k-1) & z^*(k-1) \end{bmatrix} = \begin{bmatrix} R_k(+) & z_k(+) \\ 0 & \# \end{bmatrix} \tag{23} $$

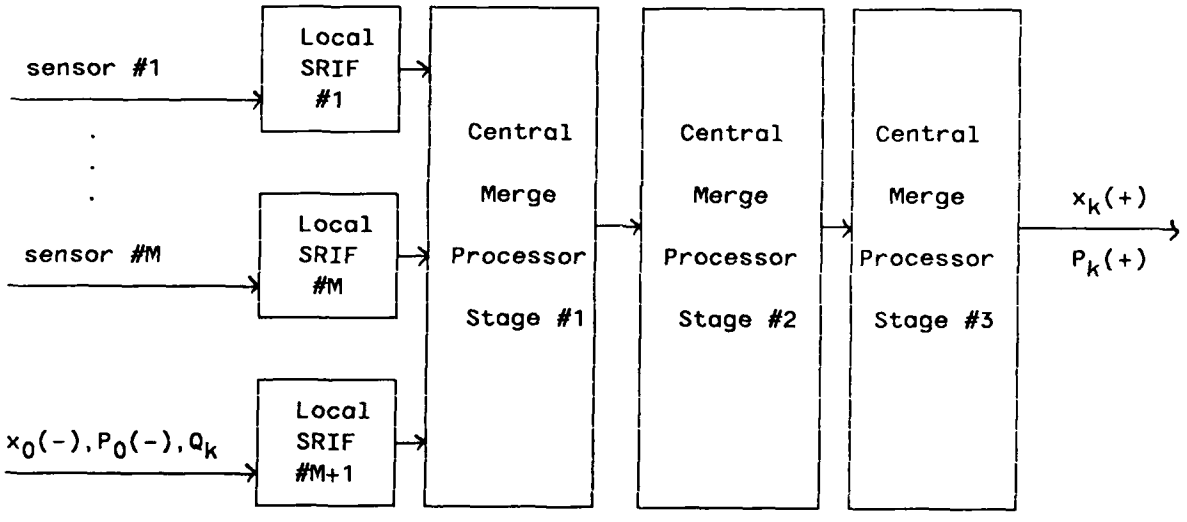
where ${}^4T_k$ is an orthogonal transformation which puts the left hand side of (23) in upper triangular form. Globally optimal filter estimates and covariances are then given by

$$x_k(+) = R_k^{-1}(+)\, z_k(+) \tag{24}$$

$$P_k(+) = R_k^{-1}(+)\, R_k^{-tr}(+) \tag{25}$$

When the a priori information about the initial condition and process noise models are adjusted, it is only necessary to rerun (21) and (23) without reprocessing any measurements.

To summarize, the measurements from each tracking sensor may be processed by a local SRIF



(15)-(20) which generates a set of smoothing coefficients $R_w^{*(i)}$, $R_{wx}^{*(i)}(+)$, $z_w^{*(i)}(+)$ as well as a square root information matrix $R(+)^{(i)}$ and information vector $z(+)^{(i)}$. The central or merge processor consists of three separate processors which operate in parallel. The first mechanizes (21),(22) which combines the local smoothing coefficients with the effects of process noise and prior information about the initial state. The second mechanizes (23) which merges the local square root information matrices and vectors with output from the first, but only upon demand by the third. The third produces

estimates and covariances whenever desired by back-solving (24) and (25) respectively, noting that (24),(25) require output from the second processor. An important observation is that feedback of information from the merge processor to the local filters is not necessary here. This helps to keep the bandwidth of communication between the local processors and merge processor from exceeding hardware limitations.

2.0 Results of the Phase I Work

In order to validate our decentralized approach to solving the linear least squares estimation problem, two tasks were performed. First, in the next section 2.1 we construct and execute a simulation of the DSRIF within an underwater test range environment. The nominal trajectory is assumed to be known a priori so that actually, perturbations to the nominal are estimated. Then, in section 2.2 we complete our processing of real Multiple Rocket Launch System Data provided by the White Sands Missile Range. In this case, an adaptive form of the extended DSRIF, wherein the process noise levels were chosen to be a function of the globally optimal estimate error covariance, was used.

2.1 Validation of the DSRIF for a Simulated Underwater Zig-Zag Maneuver

A low level simulation of an underwater multisensor network, tracking a maneuvering underwater vehicle was encoded in Fortran '77 and executed on an IBM (clone) Model AT desktop computer (640K ram, 40 Mbyte hard disk, Intel 80287 math coprocessor). Initial conditions for the nominal trajectory were calculated using a "flat earth" or constant gravitational acceleration model which neglects the earth's rotation, neglects hydrodynamic drag, and assumes that the target vehicle is a point mass.

The initial position is located within the polygon formed by connecting adjacent sensors, and the desired terminal position is 1.86 miles downrange. However, 2 instantaneous reversals of the cross range velocity at 42 and 124 seconds were made, and this resulted in a total path length of 2.62 miles when projected onto the "Range Area General Map" (RAGM)" (which we use to define the GCS). That is, x^1, x^2, x^3 is a right handed coordinate system where the vector cross product of x^1 with x^2 is equal to x^3, and the vectors $[x^1\ 0\ 0]$ and $[0\ x^2\ 0]$ point east along latitude 32.380 degrees and north along longitude 106.481 degrees respectively. Then $[0\ 0\ x^3]$ is collinear with the radial vector which points outward from the earth's center and passes through the origin of the GCS. Choosing the launch elevation angle, w.r.t. the x^1, x^2 tangent plane, to be -3.96 degrees results in a 200 ft depth variation over the path length. The corresponding initial position and velocity is

$$\begin{bmatrix} x_0^1 \\ x_0^2 \\ x_0^3 \\ x_0^6 \\ x_0^4 \\ x_0^5 \end{bmatrix} = \begin{bmatrix} 7{,}920 & \text{feet} \\ 5{,}280 & \text{feet} \\ 0 & \text{feet} \\ -4.14 & \text{feet/sec} \\ 59.74 & \text{feet/sec} \\ 59.74 & \text{feet/sec} \end{bmatrix} \tag{26}$$

and the total time to traverse the path was 167 seconds.

Five sensor locations which surround the projected flight path were selected. Each sensor records measurements with respect to its own LCS so that coordinate transformations to the GCS were derived and included in the observational equations. The transformation is

$$\begin{bmatrix} x^1 \\ x^2 \\ x^3 \end{bmatrix} = (T^i) \begin{bmatrix} x^{i,1} \\ x^{i,2} \\ x^{i,3} \end{bmatrix} + d^i \tag{27}$$

where

$$T^i = ({}^8T^i) ({}^7T^i) ({}^6T^i) ({}^5T^i) \tag{28}$$

and

$${}^5T^i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(L+\alpha^i) & \sin(L+\alpha^i) \\ 0 & -\sin(L+\alpha^i) & \cos(L+\alpha^i) \end{bmatrix} \tag{29}$$

$${}^6T^i = \begin{bmatrix} \cos\beta^i & 0 & -\sin\beta^i \\ 0 & 1 & 0 \\ \sin\beta^i & 0 & \cos\beta^i \end{bmatrix} \tag{30}$$

$${}^7T^i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos L & -\sin L \\ 0 & \sin L & \cos L \end{bmatrix} \tag{31}$$

$${}^8T^i = \begin{bmatrix} \cos\tau^i & \sin\tau^i & 0 \\ -\sin\tau^i & \cos\tau^i & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{32}$$

noting that $(T^i)^{tr} = (T^i)^{-1}$ since T^i is an orthogonal transformation. A computational savings results when $({}^5T^i)({}^6T^i)({}^7T^i)$ are combined using trigonometric identities for the cosine and sine of the sum of two angles along with the small angle approximation since α^i is bounded by $\pm.025$ degrees over the length of the underwater Test Range.

d^i Is the vector from the GCS to the ith LCS and $\alpha^i, \beta^i, \tau^i$ are the three Euler angles describing the orientation of the ith LCS w.r.t. the GCS. Specifically, if we first rotate about x^1 counterclockwise by an $(L+\alpha^i)$ degree change in latitude (aligning y^2 with the polar axis), and then rotate counterclockwise about y^2 by a β^i degree change in longitude, (β^i is bounded by $\pm.025$ degrees over the width of the Test Range) then rotate clockwise about x^3 by an L degree change in latitude, and finally rotate about z^4 by τ^i degrees to account for "tangent plane misalignment", the LCS will coincide with the GCS when $d^i = 0$. α^i Is equal to the LCS latitude - GCS latitude. β^i Is equal to the LCS longitude - GCS longitude.

To determine α^i, β^i given L, δ and d^i (in global coordinates), solve the following two equations:

$$\begin{bmatrix} {}^ox^{i,1} \\ {}^ox^{i,2} \\ {}^ox^{i,3} \end{bmatrix}_{ECEF} = \begin{bmatrix} {}^ox^1 \\ {}^ox^2 \\ {}^ox^3 \end{bmatrix}_{ECEF} + T^g \begin{bmatrix} d^{i,1} \\ d^{i,2} \\ d^{i,3} \end{bmatrix}_{GCS} \tag{33}$$

where

$$\begin{bmatrix} {}^ox^1 \\ {}^ox^2 \\ {}^ox^3 \end{bmatrix}_{ECEF} = \begin{bmatrix} r_e \cos L \cos \delta \\ -r_e \cos L \sin \delta \\ r_e \sin L \end{bmatrix} \tag{34}$$

and

$$T^g = \begin{bmatrix} \sin\delta & -\cos\delta & 0 \\ \cos\delta & \sin\delta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \sin L & -\cos L \\ 0 & \cos L & \sin L \end{bmatrix} \tag{35}$$

$$\begin{bmatrix} {}^ox^{i,1} \\ {}^ox^{i,2} \\ {}^ox^{i,3} \end{bmatrix}_{ECEF} = \begin{bmatrix} {}^or^i \cos(L+\alpha^i) \cos(\delta+\beta^i) \\ {}^or^i \cos(L+\alpha^i) \sin(\delta+\beta^i) \\ {}^or^i \sin(L+\alpha^i) \end{bmatrix} \tag{36}$$

The solution is

$$L+\alpha^i = \tan^{-1} \frac{({}^ox^{i,3})^2}{({}^ox^{i,1})^2 + ({}^ox^{i,2})^2} \qquad (L+\alpha^i) \approx 32 \text{ degrees} \tag{37}$$

$$\delta+\beta^i = \cos^{-1} \frac{{}_o x^{i,1}}{{}_o r^i \cos(L+\alpha^i)} \qquad (\delta+\beta^i) \approx 106 \text{ degrees} \tag{38}$$

where

$${}_o r^i = \frac{{}_o x^{i,3}}{\sin(L+\alpha^i)} \tag{39}$$

The results are given in Table 1 below.

i	Sensor Type	$d^{i,1}$	$d^{i,2}$	$d^{i,3}$	$L+\alpha^i$	$\delta+\beta^i$	τ^i
1	ps	0.50	0.50	0.	32.3872	106.4724	0.
2	ps	0.25	3.00	0.	32.4233	106.4767	0.
3	as	2.00	4.00	0.	32.4378	106.4467	0.
4	ps	2.50	2.00	0.	32.4089	106.4382	0.
5	as	3.00	1.00	0.	32.3944	106.4296	0.

Table 1: Passive Sonar (ps) and Active Sonar (as) Locations and Orientations in GCS for the Underwater Target

Detailed equations that describe the translational motion of the target were developed. The equations include a radial gravitational force as well as centrifugal and Coriolis forces which come about by rotation of the GCS about the polar axis. The equations are

$$\dot{x}^4 = \frac{-\mu x^1}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{3/2}} + 2\Omega(x^5 \sin L - x^6 \cos L) + \Omega^2 x^1 \tag{40}$$

$$\dot{x}^5 = \frac{-\mu x^2}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{3/2}} - 2\Omega x^4 \sin L + \Omega^2(x^2 \sin^2 L - (x^3+r_e)\cos L \sin L) \tag{41}$$

$$\dot{x}^6 = \frac{-\mu(x^3+r_e)}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{3/2}} + 2\Omega x^4 \cos L + \Omega^2((x^3+r_e)\cos^2 L - x^2 \cos L \sin L) \tag{42}$$

where

$$\mu = G\, m_e \tag{43}$$

and all other variables are defined in the List of Symbols. A spherical harmonic expansion of the earth's gravitational field is not necessary since the variation in target depth was small relative to the earth's radius. However, a major effect is hydrodynamic drag and buoyancy which we accounted for by reducing G to 1/1000 of its correct value. Thus, the trajectories tended to be unstable (due to centrifugal forces dominating) but reasonable ones could be obtained by careful tuning of the initial condition. More precise modeling will be undertaken in Phase II.

Any one of three sensor types may reside at each sensor location. The three types are

- active sonar wherein range (r^i), azimuth (Γ^i) and elevation (θ^i) data are available,
- passive sonar wherein azimuth and elevation data are available, and
- doppler sonar wherein range, range rate ($\dot{r}^i$), azimuth and elevation data are available.

The observational equations in terms of the global state are

$$r^i = ((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2} \tag{44}$$

$$\theta^i = \sin^{-1} \frac{[0\ 0\ 1]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \tag{45}$$

$$\Gamma^i = \tan^{-1} \frac{[0\ 1\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix}}{[1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix}} \tag{46}$$

$$\dot{r}^i = \frac{[\ x^4\ \ x^5\ \ x^6\ ] \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \tag{47}$$

The various measurement variables are defined by Figure 1 that follows.

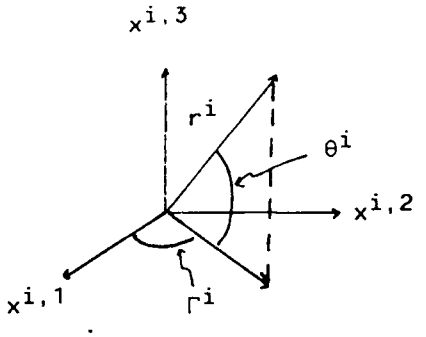


Figure 1: The Local Coordinate System and Measurement Variables

The nonlinear equations of motion (40)-(42) were integrated forwards in time starting from (26) and using the Fourth Order Runge Kutta method with a 1 second interval. Figures 2 and 3 show the resultant trajectory of the target vehicle. As a further check on the Fortran code for this part of the simulation, the total translational energy (kinetic plus potential) of the target was computed for each point along the computed trajectory. The total energy remained constant to within 1% as it should since there are no external forces ($u(t) = 0$) and the system is conservative. Figures 4 through 8 show some of the corresponding measurements for the various data-types with $v^i = 0$ for $\{1 \le i \le M\}$.

Each row of A(t) is computed by partial differentiating the same row of f(x(t)) w.r.t. the nominal state. Thus, A(t) has the following structure

$$A(t) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{\partial \dot{x}^4}{\partial x^1} & & \cdots & & & \frac{\partial \dot{x}^4}{\partial x^6} \\ \vdots & & & & & \vdots \\ \frac{\partial \dot{x}^6}{\partial x^1} & & \cdots & & & \frac{\partial \dot{x}^6}{\partial x^6} \end{bmatrix} \qquad (48)$$

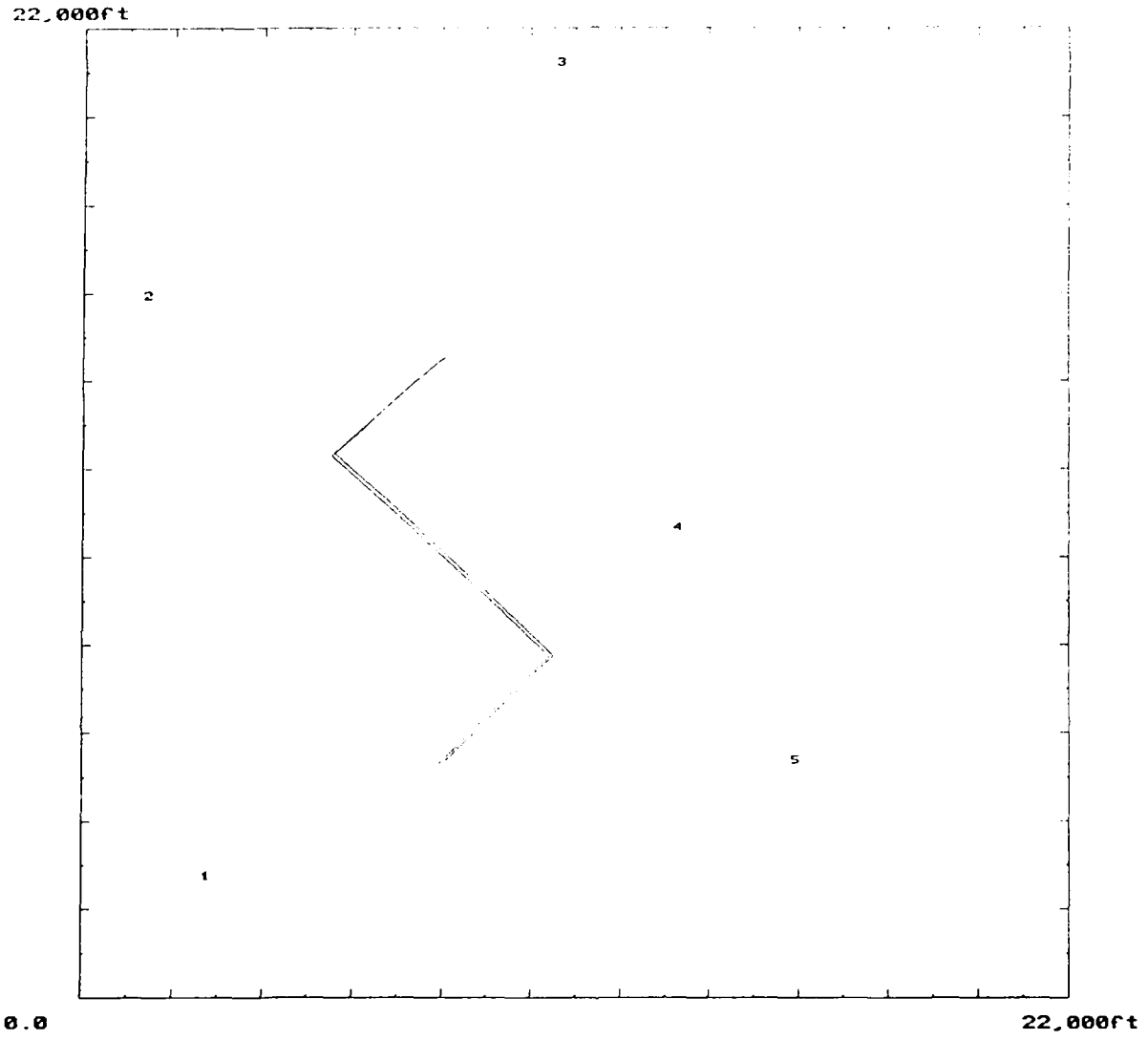


Figure 2

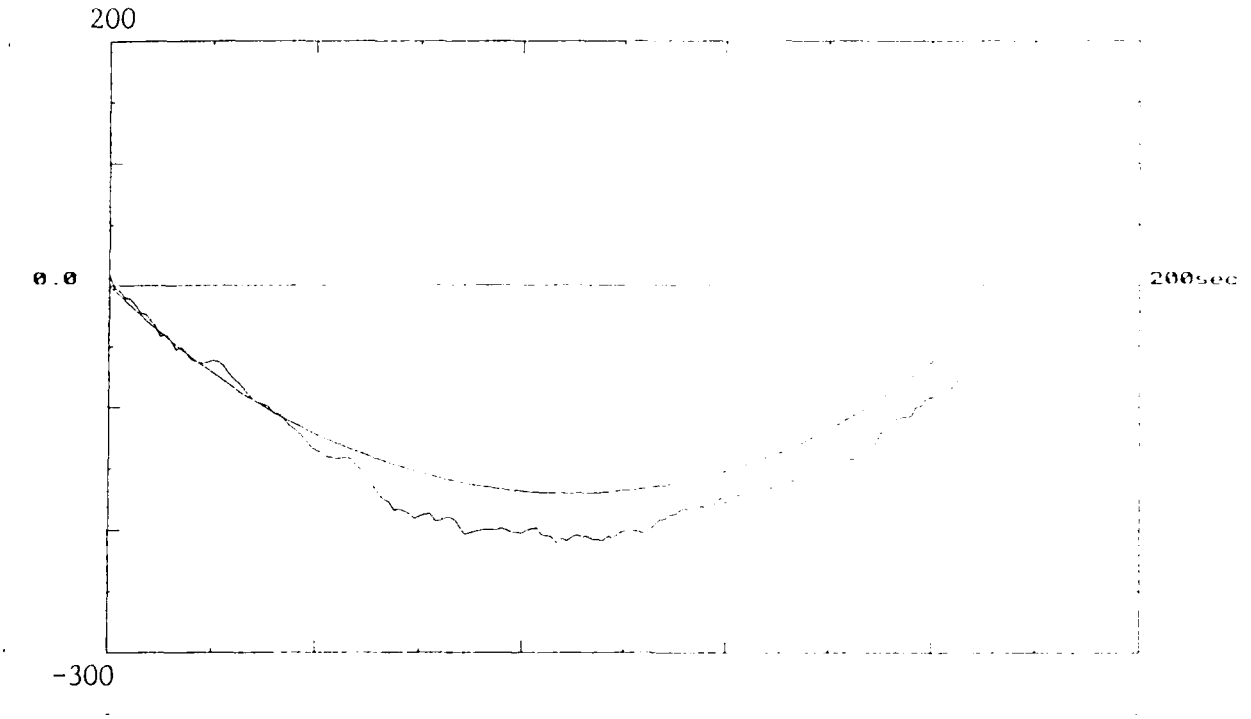


Figure 3

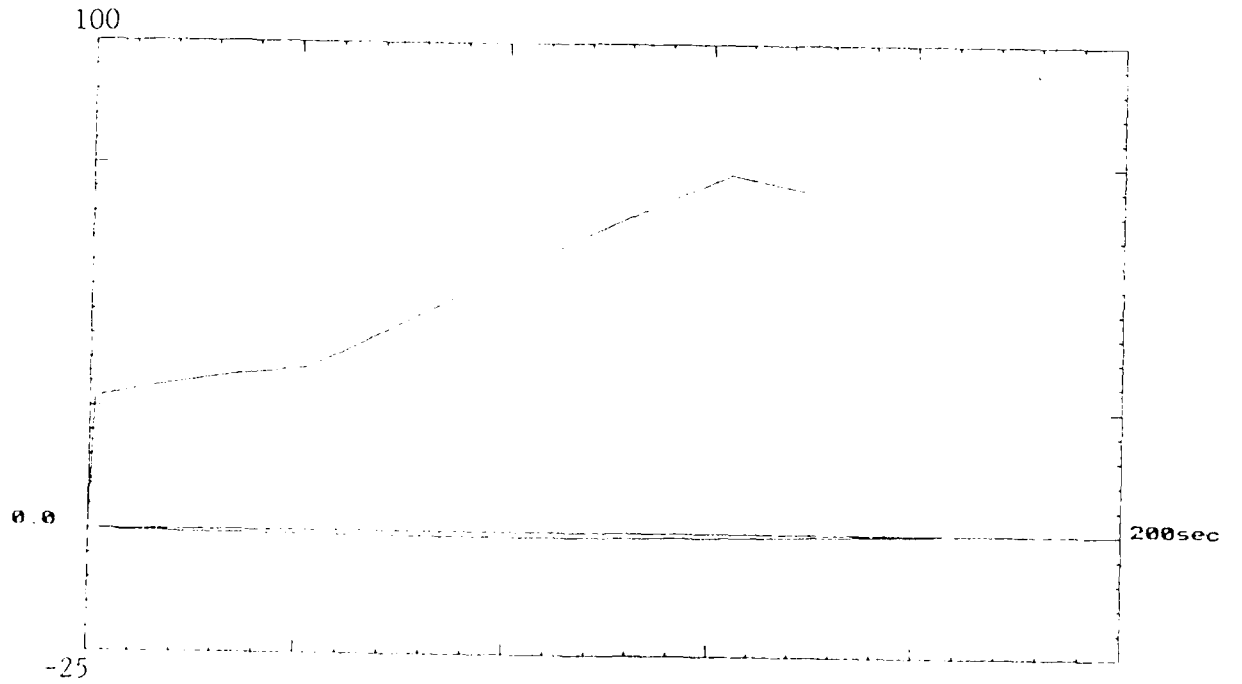


Figure 4

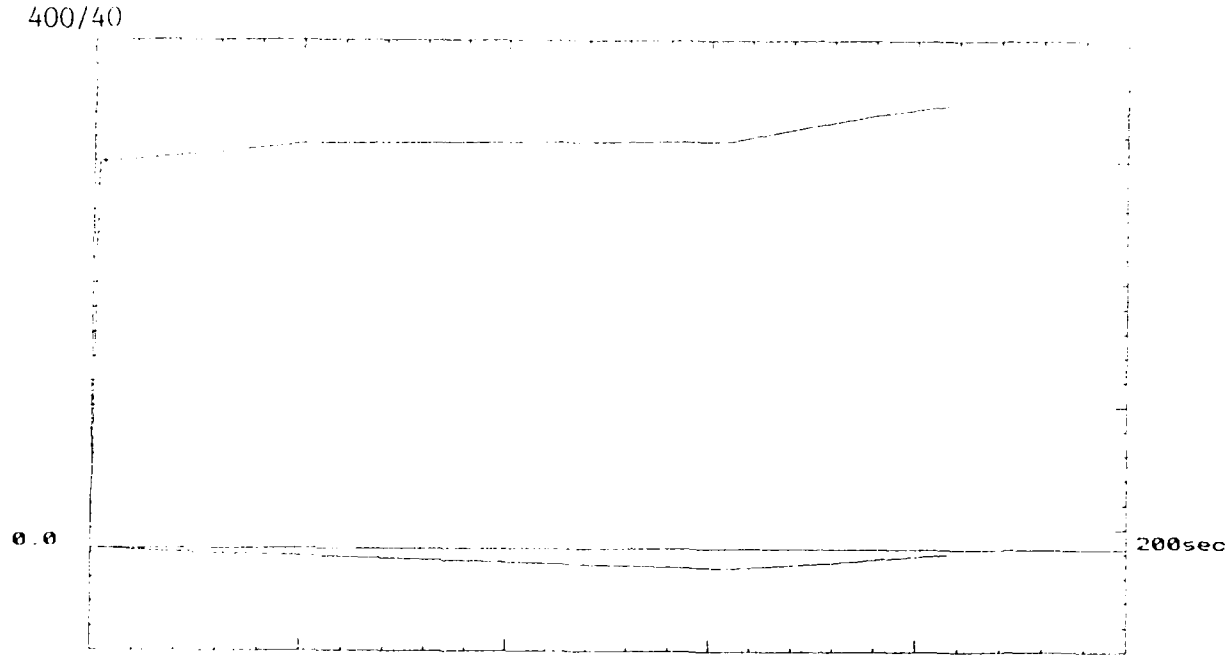


Figure 5

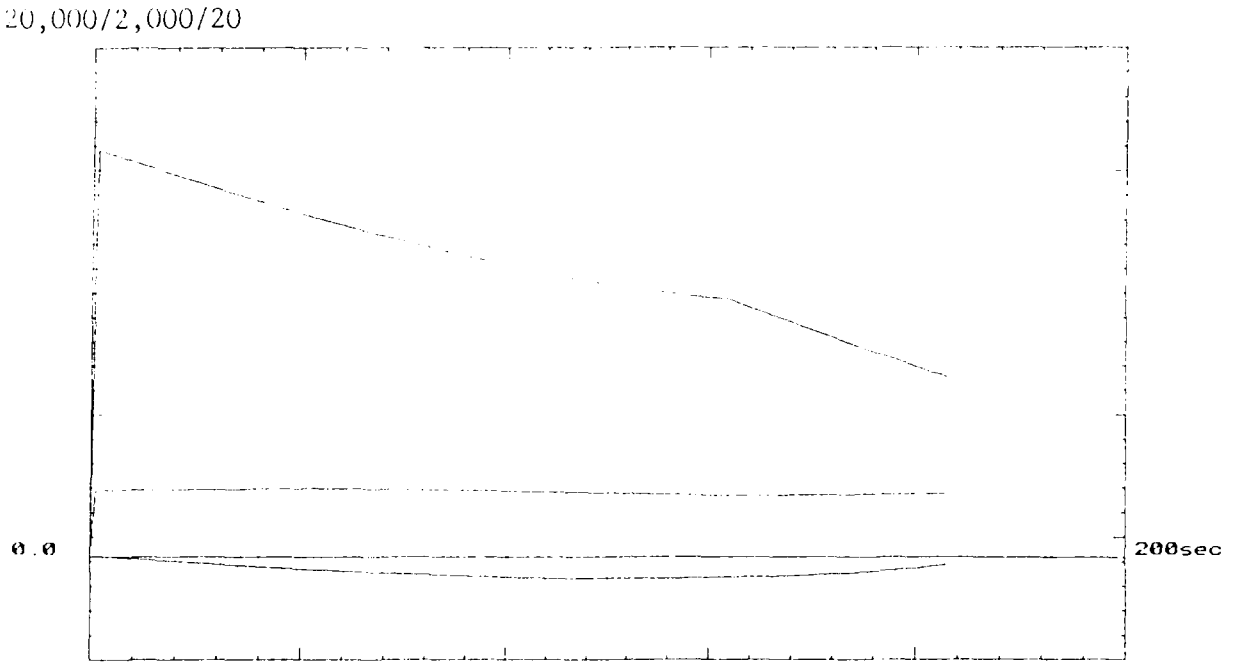


Figure 6

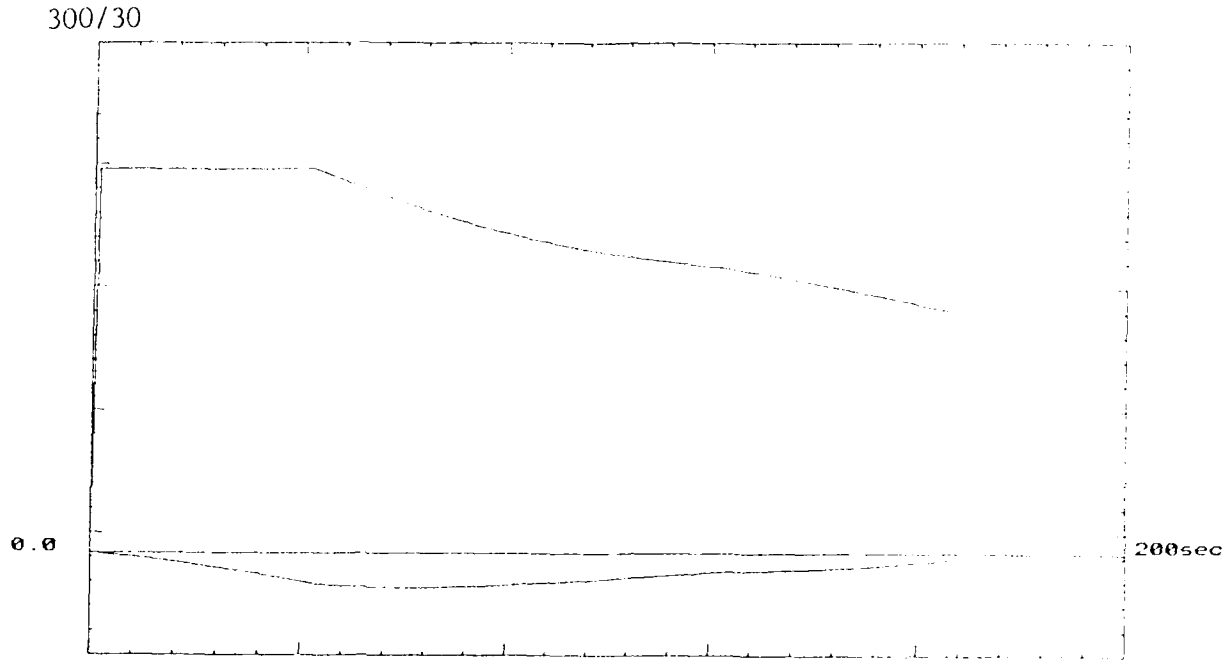


Figure 7

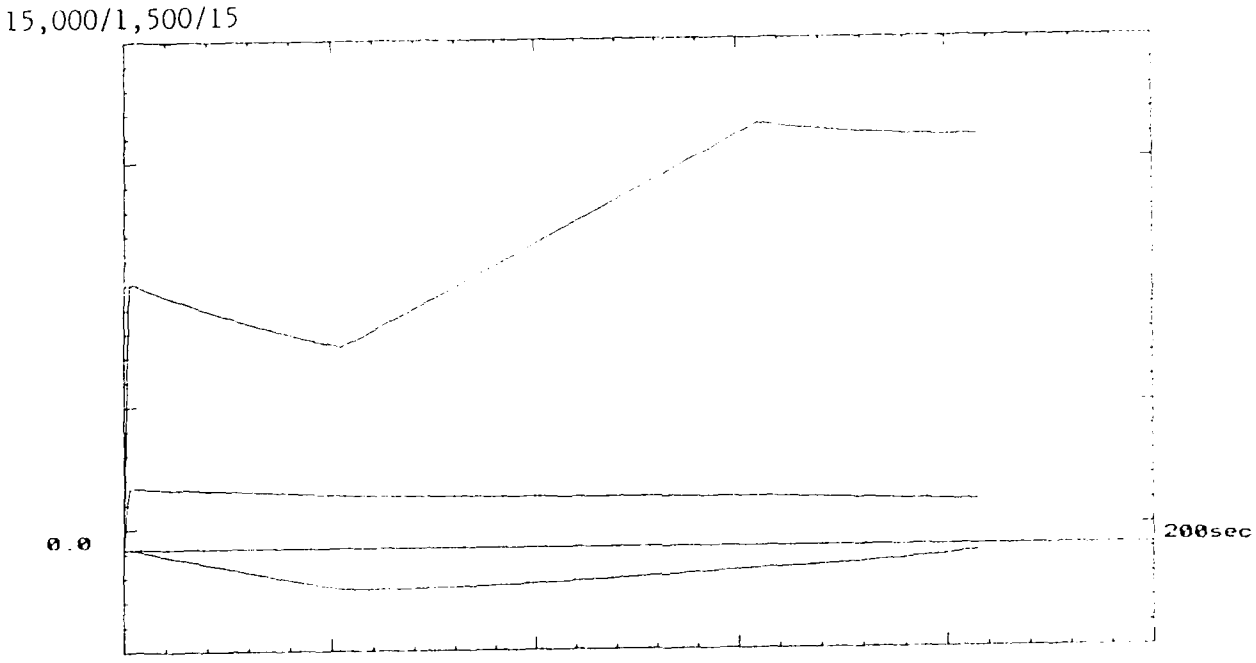


Figure 8

where

$$\frac{\partial \dot{x}^4}{\partial x^1} = \frac{-\mu}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{3/2}} + \frac{3\mu(x^1)^2}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} + \Omega^2 \tag{49}$$

$$\frac{\partial \dot{x}^4}{\partial x^2} = \frac{3\mu x^1 x^2}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} \tag{50}$$

$$\frac{\partial \dot{x}^4}{\partial x^3} = \frac{3\mu x^1 (x^3+r_e)}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} \tag{51}$$

$$\frac{\partial \dot{x}^4}{\partial x^4} = 0 \tag{52}$$

$$\frac{\partial \dot{x}^4}{\partial x^5} = 2\Omega \sin L \tag{53}$$

$$\frac{\partial \dot{x}^4}{\partial x^6} = -2\Omega \cos L \tag{54}$$

$$\frac{\partial \dot{x}^5}{\partial x^1} = \frac{3\mu x^1 x^2}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} \tag{55}$$

$$\frac{\partial \dot{x}^5}{\partial x^2} = \frac{-\mu}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{3/2}} + \frac{3\mu(x^2)^2}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} + \Omega^2 \sin^2 L \tag{56}$$

$$\frac{\partial \dot{x}^5}{\partial x^3} = \frac{3\mu x^2 (x^3+r_e)}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} - \Omega^2 \cos L \sin L \tag{57}$$

$$\frac{\partial \dot{x}^5}{\partial x^4} = -2\Omega \sin L \tag{58}$$

$$\frac{\partial \dot{x}^5}{\partial x^5} = 0 \tag{59}$$

$$\frac{\partial \dot{x}^5}{\partial x^6} = 0 \tag{60}$$

$$\frac{\partial \dot{x}^6}{\partial x^1} = \frac{3\mu x^1 (x^3+r_e)}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} \tag{61}$$

$$\frac{\partial \dot{x}^6}{\partial x^2} = \frac{3\mu x^2 (x^3+r_e)}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} - \Omega^2 \cos L \sin L \tag{62}$$

$$\frac{\partial \dot{x}^6}{\partial x^3} = \frac{-\mu}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{3/2}} + \frac{3\mu (x^3+r_e)^2}{((x^1)^2 + (x^2)^2 + (x^3+r_e)^2)^{5/2}} + \Omega^2 \cos^2 L \tag{63}$$

$$\frac{\partial \dot{x}^6}{\partial x^4} = 2\Omega \cos L \tag{64}$$

$$\frac{\partial \dot{x}^6}{\partial x^5} = 0 \tag{65}$$

$$\frac{\partial \dot{x}^6}{\partial x^6} = 0 \tag{66}$$

Each row of $C^i(t)$ is computed by partial differentiating the same row of $h_i(x(t))$ w.r.t. the nominal state. Thus, when the ith sensor is a doppler sonar, $C^i(t)$ has the following structure

$$C^i(t) = \begin{bmatrix} \frac{\partial r^i}{\partial x^1} & \cdots & \frac{\partial r^i}{\partial x^3} & 0 & 0 & 0 \\ \frac{\partial \dot{r}^i}{\partial x^1} & & & & & \frac{\partial \dot{r}^i}{\partial x^6} \\ \frac{\partial r^i}{\partial x^1} & \cdots & \frac{\partial r^i}{\partial x^3} & 0 & 0 & 0 \\ \frac{\partial \theta^i}{\partial x^1} & \cdots & \frac{\partial \theta^i}{\partial x^3} & 0 & 0 & 0 \end{bmatrix} \quad (67)$$

where

$$\frac{\partial r^i}{\partial x^1} = \frac{x^1 - d^{i,1}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \quad (68)$$

$$\frac{\partial r^i}{\partial x^2} = \frac{x^2 - d^{i,2}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \quad (69)$$

$$\frac{\partial r^i}{\partial x^3} = \frac{x^3 - d^{i,3}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \quad (70)$$

$$\frac{\partial ri}{\partial x^4} = \frac{\partial ri}{\partial x^5} = \frac{\partial ri}{\partial x^6} = 0 \quad (71)$$

$$\frac{\partial \dot{r}i}{\partial x^1} = - \frac{(x^4(x^1-d^{i,1}) + x^5(x^2-d^{i,2}) + x^6(x^3-d^{i,3}))(x^1-d^{i,1})}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{3/2}} \quad (72)$$

$$+ \frac{x^4}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \quad (73)$$

$$\frac{\partial \dot{r}^i}{\partial x^2} = -\frac{(x^4(x^1-d^{i,1}) + x^5(x^2-d^{i,2}) + x^6(x^3-d^{i,3}))(x^2-d^{i,2})}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{3/2}} + \frac{x^5}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \tag{74}$$

$$\frac{\partial \dot{r}^i}{\partial x^3} = -\frac{(x^4(x^1-d^{i,1}) + x^5(x^2-d^{i,2}) + x^6(x^3-d^{i,3}))(x^3-d^{i,3})}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{3/2}} + \frac{x^6}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \tag{75}$$

$$\frac{\partial \dot{r}^i}{\partial x^4} = \frac{x^1-d^{i,1}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \tag{76}$$

$$\frac{\partial \dot{r}^i}{\partial x^5} = \frac{x^2-d^{i,2}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \tag{77}$$

$$\frac{\partial \dot{r}^i}{\partial x^6} = \frac{x^3-d^{i,3}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \tag{78}$$

$$\frac{\partial \theta^i}{\partial x^1} = \frac{1}{(1-u^2)^{1/2}} \left(-\frac{[0\ 0\ 1]\,(T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} (x^1-d^{i,1})}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{3/2}} + \frac{[0\ 0\ 1]\,(T^i)^{tr} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \right) \tag{79}$$

where u is the argument of $\sin^{-1}$ in equation (45).

$$\frac{\partial \theta^i}{\partial x^2} = \frac{1}{(1-u^2)^{1/2}} - \frac{[0\ 0\ 1]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} (x^2-d^{i,2})}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{3/2}}$$

$$+ \frac{[0\ 0\ 1]\ (T^i)^{tr} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \qquad (80)$$

$$\frac{\partial \theta^i}{\partial x^3} = \frac{1}{(1-u^2)^{1/2}} - \frac{[0\ 0\ 1]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} (x^3-d^{i,3})}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{3/2}}$$

$$+ \frac{[0\ 0\ 1]\ (T^i)^{tr} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}{((x^1-d^{i,1})^2 + (x^2-d^{i,2})^2 + (x^3-d^{i,3})^2)^{1/2}} \qquad (81)$$

$$\frac{\partial \theta^i}{\partial x^4} = \frac{\partial \theta^i}{\partial x^5} = \frac{\partial \theta^i}{\partial x^6} = 0 \qquad (82)$$

$$\frac{\partial \Gamma^i}{\partial x^1} = \frac{1}{(1+u^2)} - \frac{[0\ 1\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} [1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}{\left([1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} \right)^2}$$

$$+ \frac{[0\ 1\ 0]\ (T^i)^{tr} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}{[1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix}} \qquad (83)$$

where u is now the argument of $\tan^{-1}$ in equation (46).

$$\frac{\partial r^i}{\partial x^2} = \frac{1}{(1+u^2)} \left[- \frac{[0\ 1\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} [1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}}{\left([1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} \right)^2} \right.$$

$$\left. + \frac{[0\ 1\ 0]\ (T^i)^{tr} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}}{[1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix}} \right] \qquad (84)$$

$$\frac{\partial r^i}{\partial x^3} = \frac{1}{(1+u^2)} \left[- \frac{[0\ 1\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} [1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}{\left([1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1-d^{i,1} \\ x^2-d^{i,2} \\ x^3-d^{i,3} \end{bmatrix} \right)^2} \right.$$

$$+ \frac{[0\ 1\ 0]\ (T^i)^{tr} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}{[1\ 0\ 0]\ (T^i)^{tr} \begin{bmatrix} x^1 - d^{i,1} \\ x^2 - d^{i,2} \\ x^3 - d^{i,3} \end{bmatrix}} \qquad (85)$$

$$\frac{\partial r^i}{\partial x^4} = \frac{\partial r^i}{\partial x^5} = \frac{\partial r^i}{\partial x^6} = 0 \qquad (86)$$

The 6 biases $\alpha^i, \beta^i, \tau^i, d^{i,1}, d^{i,2}, d^{i,3}$ per local system could be included as states in the filter and estimated in order to correct for any preflight miscalibration. Then, $\alpha^i = 0$, $\beta^i = 0$ ... would be added to the equations of motion and additional partial derivative expressions would be needed. However, only the 6 positions and velocities of the underwater target were included.

In order to create simulated data and test the state estimation part of the code, a subroutine RANDOM for generating sequences of Gaussian random vectors with prescribed covariance was used. Two algorithms were considered in the derivation of the subroutine. The first proceeds by rotating coordinates to a system in which the covariance matrix is diagonal. In this system the multivariate normal density becomes equal to the product of its marginal densities, and each marginal density can be sampled independently of the other components. After obtaining a sample vector in this rotated system, the coordinates are rotated back to the original system.

The second algorithm proceeds by decomposing the multivariate normal density into the product of the marginal density of the first variate times the joint density of the remaining variates, conditional upon the value sampled for the first. This joint density is determined once the first variate has been sampled from its marginal density. The procedure is then applied to the second variate and iterated until values have been assigned to all components of the sample vector. This "Conditional Decomposition Algorithm" will execute more rapidly than the latter "Matrix Diagonalization Algorithm" especially for time varying covariance matrices. Thus it was chosen as the basis for subroutine RANDOM.

Assuming a constant covariance matrix, RANDOM has been tested by counting the number of random values within several bands for each component. Comparison with theory has shown agreement to within a few percent.

Figures 9 and 10 show the evolution of the perturbed state as governed by equation (3). The initial condition is

$$\delta x_0 = [\ 79.2\text{ft} \quad 0.\text{ft} \quad 0.\text{ft} \quad 0.\text{ft/s} \quad 0.\text{ft/s} \quad 0.\text{ft/s}\ ]^{tr}$$

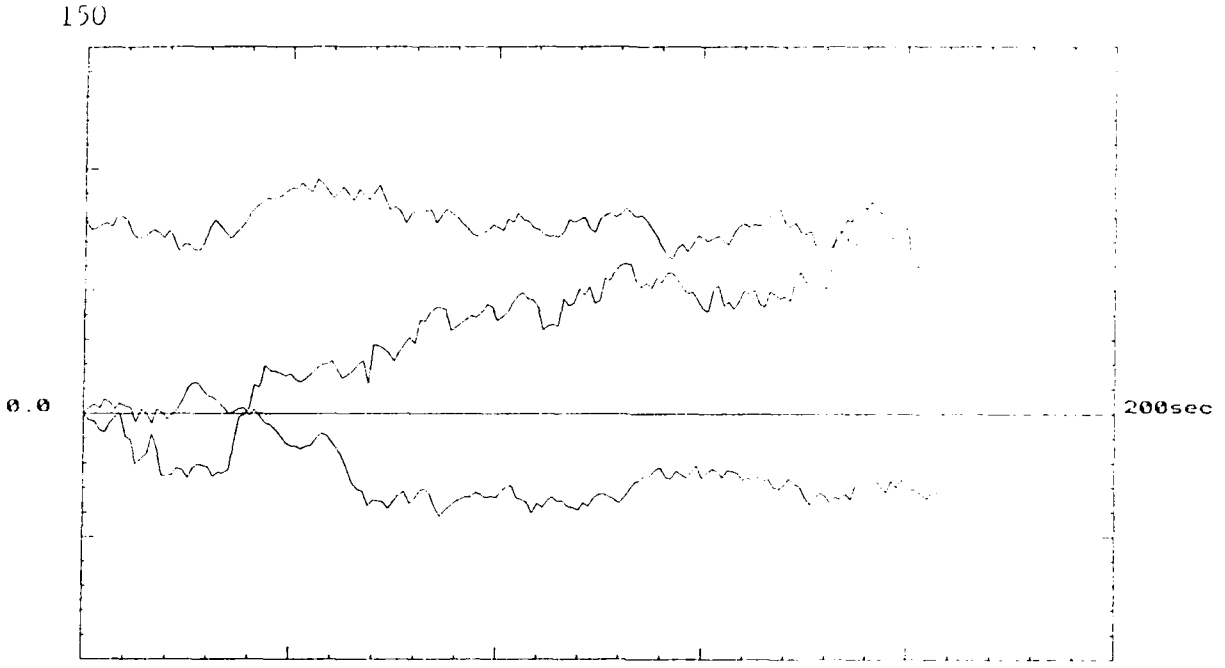


Figure 9

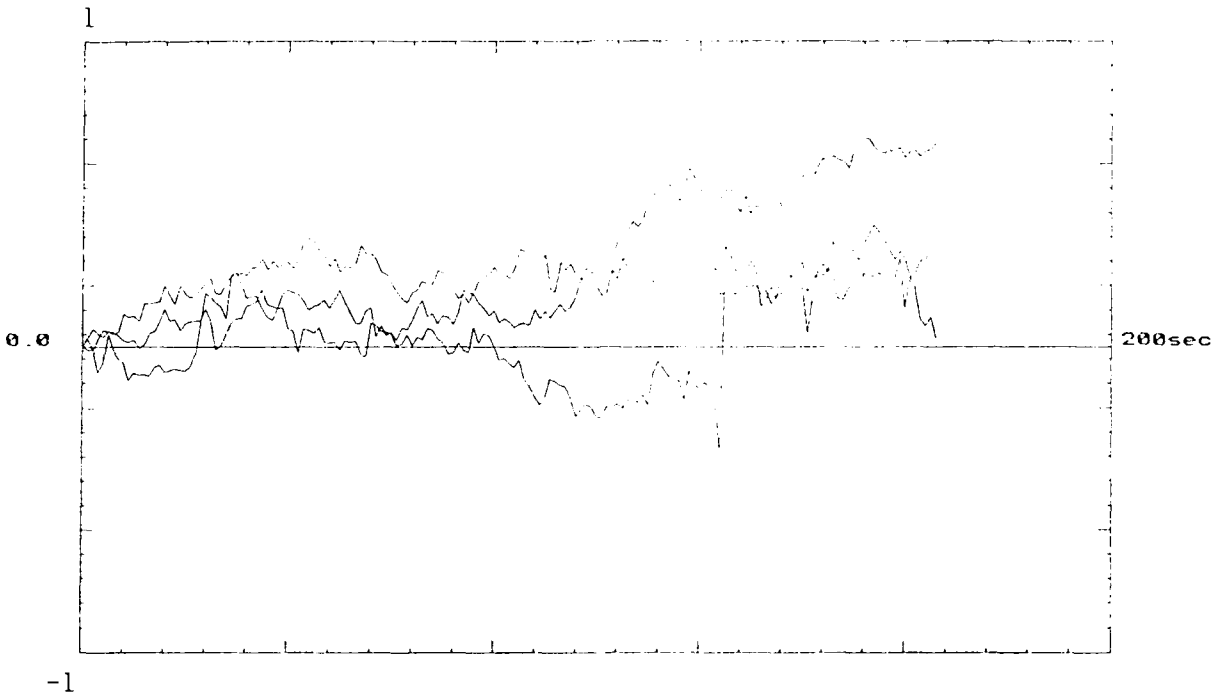


Figure 10

which represents a perturbation in the x location only. However, the resulting perturbed trajectories in the y and z directions are nonzero due to cross coupling terms within A(t). A diagonal Q_k with small variances was used to generate the process noise sequence. The position variances were 10 ft^2 and the velocity variances were 10^{-3} $(ft/s)^2$.

Figures 11, 12 and 13 are the corresponding DSRIF results with all of the prior and process noise information embedded in the merge processor. Each of the 12 local filters processed 1 measurement variable. A diagonal R_k, with variances of 10^{-8} deg^2 and 1 ft^2 for angular variables and range variables respectively, was used to generate the measurement noise sequence. The initial state estimate for all of the local filters was

$$\delta x_0^i = [179.2ft \quad 100.ft \quad 10.ft \quad 5.97\ ft/s \quad 5.97\ ft/s \quad -.41\ ft/s\ ]^{tr}$$

and a diagonal $P_0(+)$, with variances of 1 ft^2 and 1 $(ft/s)^2$ for position variables and velocity variables respectively, was used to initialize the merge processor. Figures 11 and 12 show that the rms position estimate errors quickly decay to steady state mean values after only a few time samples however, the velocity estimate errors reach steady state after approximately 40 samples. The corresponding estimate error covariances in Figure 13 follow the same course as expected.

In Figure 14 the process noise levels were multiplied by 5 and 100 for position and velocity variances respectively. Comparison with Figure 13 shows that the corresponding estimate error covariances increase as well. This is as expected since Q_k is linearly related to the time updated estimate error covariance i.e., the conventional covariance time update equation is given by

$$P_{k+1}(-) = \Phi_k P_k(+) \Phi_k^{tr} + Q_k \tag{87}$$

Furthermore, the same phenomenon results when R_k is multiplied by a factor of 10^8 for angular variables and 10 for range variables in Figure 15. The conventional covariance measurement update equation in Josephson Stabilized form

$$P_k(+) = [I - K_k H_k]\, P_k(-)\, [I - K_k H_k]^{tr} + K_k R_k K_k^{tr} \tag{88}$$

may be combined with the Kalman gain equation

$$K_k = P_k(-) H_k^{tr}\, [H_k P_k(-) H_k^{tr} + R_k]^{-1} \tag{89}$$

to show that the time updated estimate error covariance is linearly related to R_k as well. The result is that

$$P_k^{-1}(+) = P_k^{-1}(-) + H_k^{tr}\, R_k^{-1}\, H_k \tag{90}$$

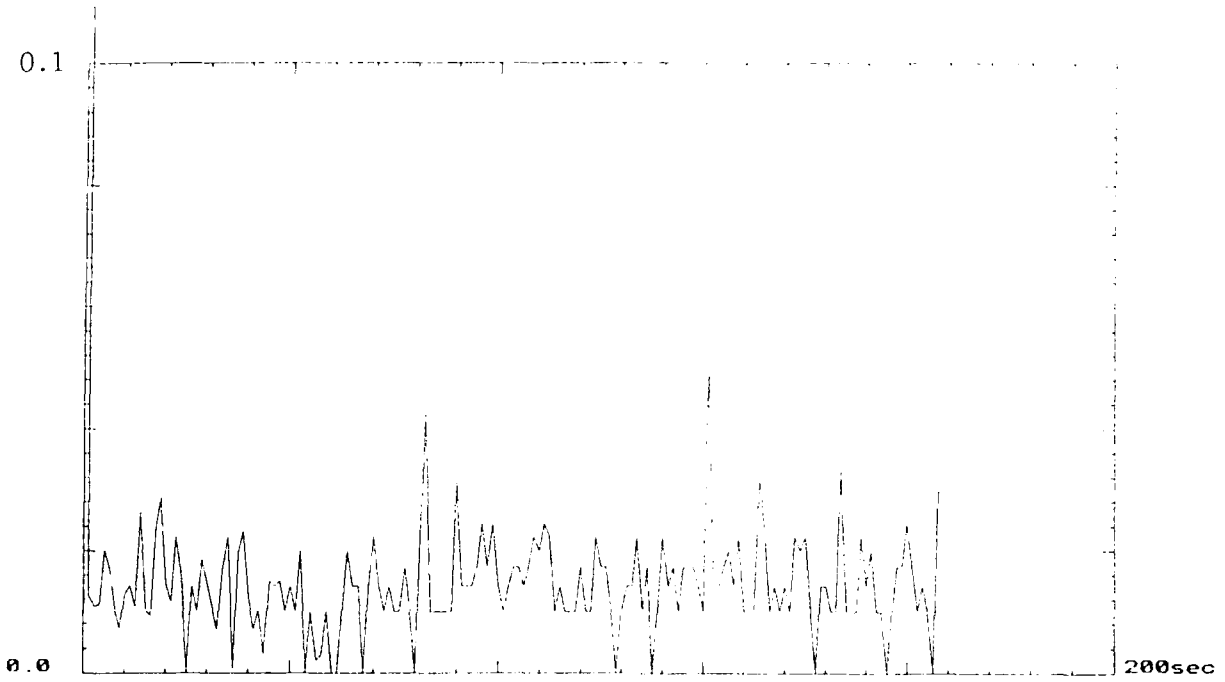


Figure 11

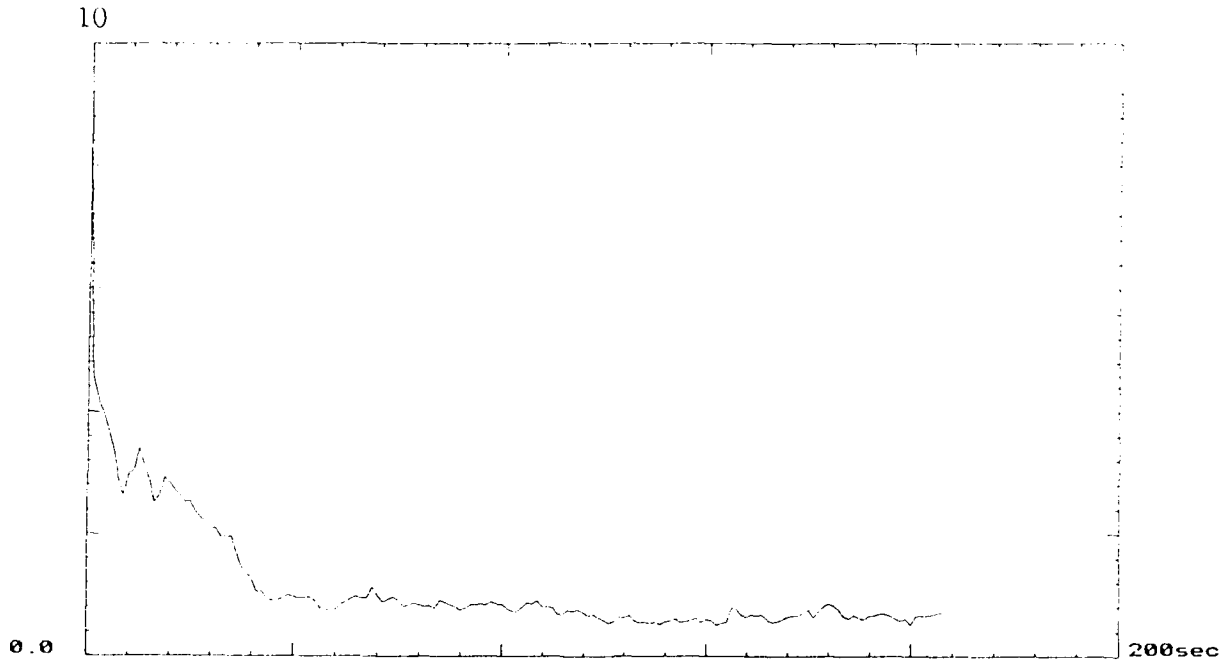


Figure 12

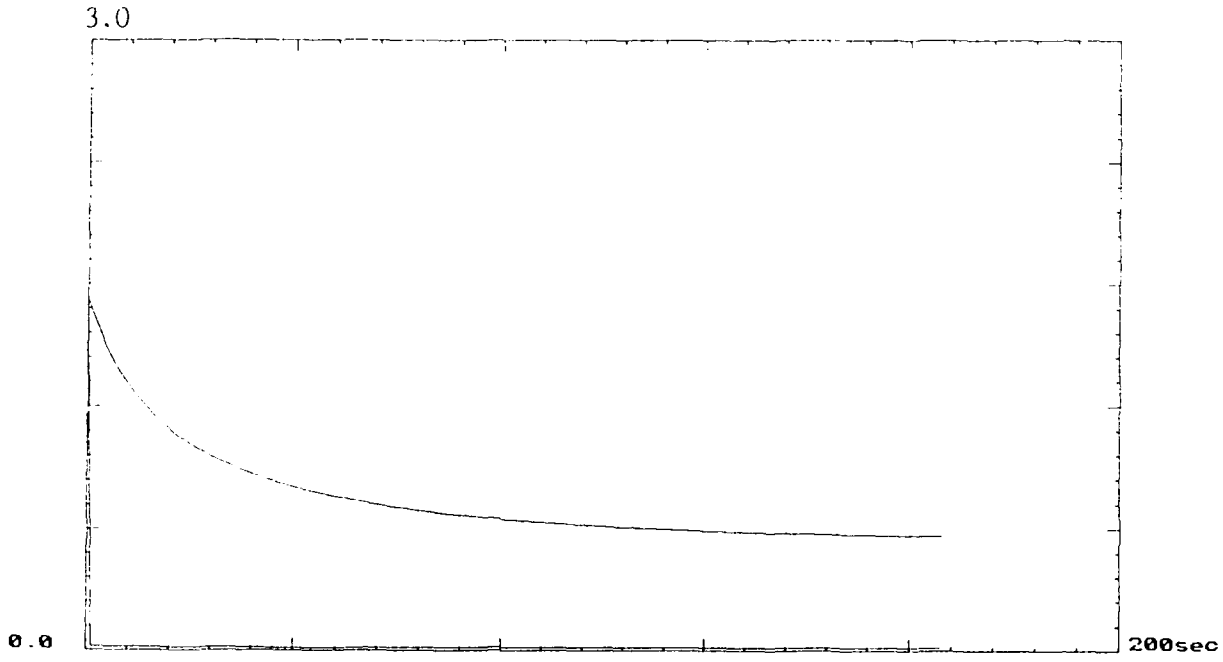


Figure 13

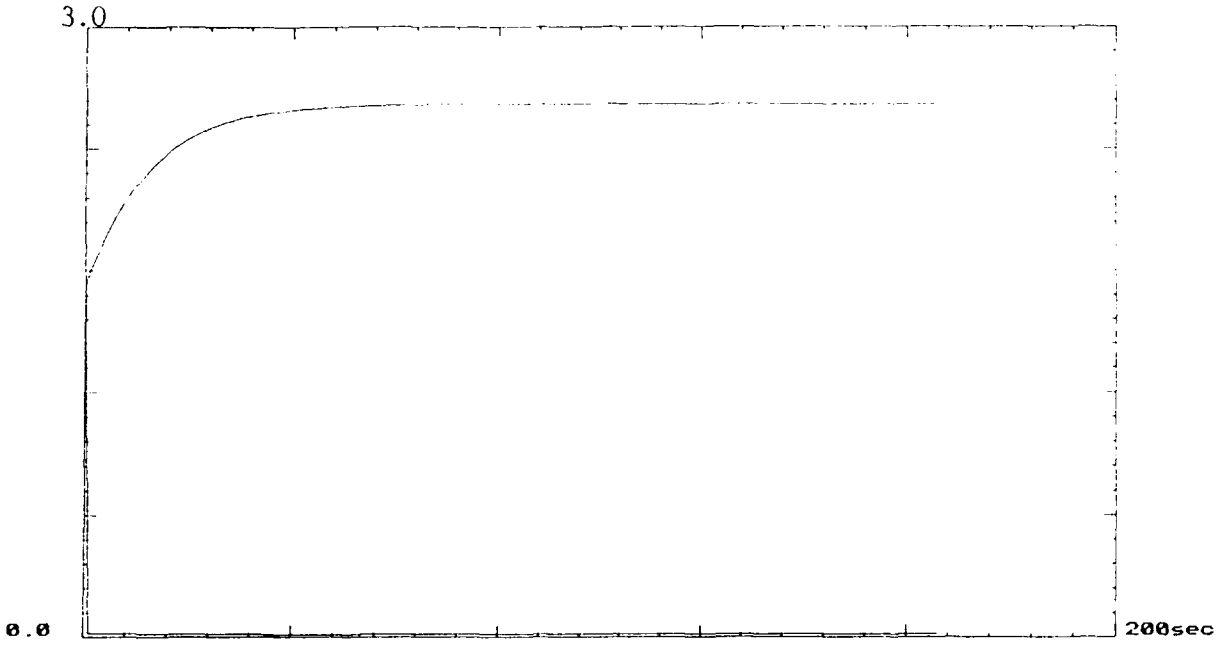


Figure 14

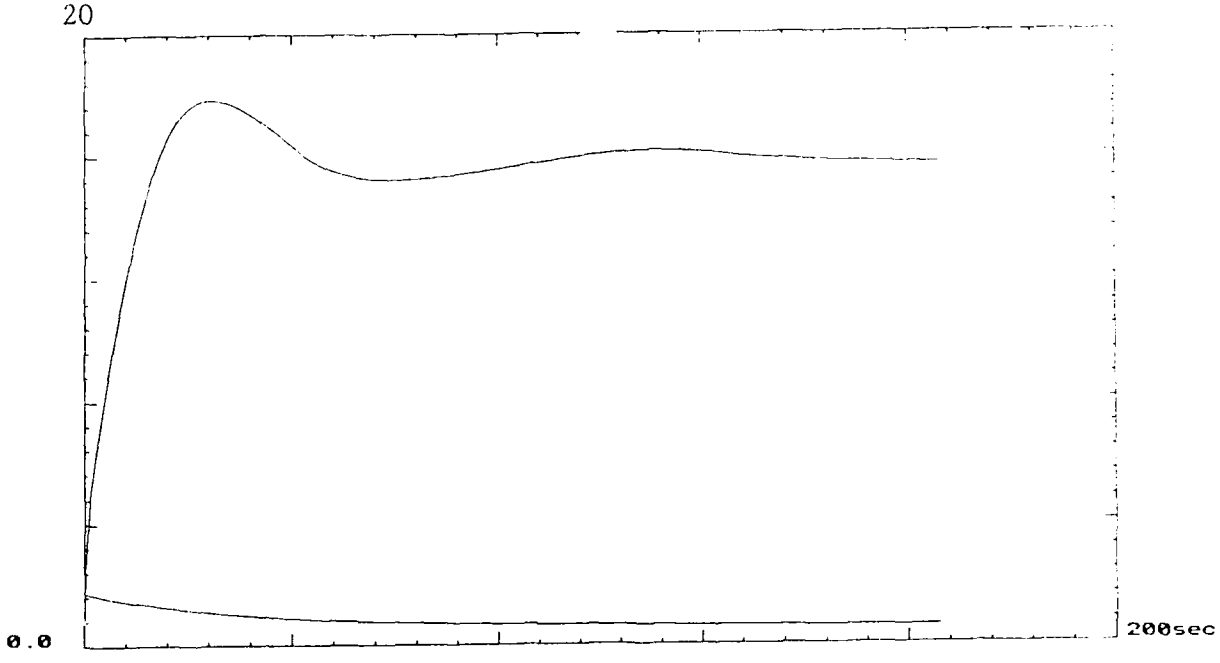


Figure 15

2.2 Extended-Decentralized Square Root Information Filtering of MLRS Data

Under SBIR Phase I contract DAAD07-87-C-0103 with the U.S. Army, White Sands Missile Range (WSMR), MTI previously derived an extended version of the algorithm and successfully used it to track real Multiple Launch Rocket System (MLRS) data provided by WSMR. In this section we describe some of the results obtained along with points of departure for future work that was needed. Then, in the next section 2.2.1 we finish processing the MLRS data.

On November 11, 1987 six rockets were launched sequentially in time over a period of 2 hours and 30 minutes at WSMR. Only 1 rocket was airborne at any one time and thus data association for multitarget tracking was not needed. The digitized measurements for all six shots were plotted in order to select the best set as characterized by the least amount of data drop outs and outlyers.

The MLRS data set contained azimuth and elevation angle measurements (with respect to each local sensor) from 11 optical trackers (analogous to passive hydrophones). The origin of the White Sands Coordinate System is at 32.38 degrees latitude and 106.481 degrees longitude. The data set also contained range, azimuth and elevation angle measurements from 3 radars (analogous to active sonar) but with respect to the local coordinate system originating at the launcher. The precise location and orientation of each sensor was known a priori and no attempt was made to estimate its uncertainty.

In this case, an extended version of the DSRIF is needed since the nominal rocket trajectories were unavailable. The same is true for underwater vehicle tracking although the velocities and accelerations are much smaller. The Extended DSRIF (E-DSRIF) may be derived by extending the observations equation, linearized about the current estimate, to

$$y_k^i = H_k^i x_k + v_k^i + z_k^i \tag{91}$$

where

$$z_k^i = h^i(x)\Big|_{x=x_k(-)} - H_k^i x_k(-) \tag{92}$$

and

$$H_k^i = \frac{\partial h^i(x)}{\partial x}\Bigg|_{x=x_k(-)}, \tag{93}$$

and extending the dynamics equation, linearized about the current estimate, to

$$x_{k+1} = F_k x_k + B_k w_k + g_k \tag{94}$$

where

$$g_k = f(x)\Big|_{x=x_k(+)} - F_k\, x_k(+) \tag{95}$$

and

$$F_k = \frac{\partial f(x)}{\partial x}\Bigg|_{x=x_k(+)} \tag{96}$$

Substituting equations (91) and (92) into the least squares performance functional of equation (13) in our DARPA Phase I proposal gives the result (details were provided in [2]).

Processing on the global scale is the same as for the DSRIF i.e., the merge steps are exactly as defined in the Phase I proposal. Only processing on the local scale is different. A major difference between the E-DSRIF and DSRIF is that the local E-DSRIFilters require knowledge of the globally optimal estimate $x_k(\pm)$ in order to compute their first order Taylor series expansion terms F_k, g_k, H_k^i and z_k^i whereas the DSRIF may compute $x_k(\pm)$ at any rate less than the highest data rate. Future research should examine whether an E-DSRIF algorithm, in which the Taylor series expansions are about the locally optimal estimates, may be derived.

In order to derive a suitable dynamical model as well as initialize the filter, the position, velocity, acceleration and jerk of the rocket were precomputed using finite differencing with $\Delta t = .1$ seconds. Results were plotted in Figures 29 through 30 (of [2]) using all of the data provided for radar #350 except for the first 21 samples (we estimated that all radar trackers or rt's had locked onto the target by the 22nd sample). Figure 30 indicated that jerk could be suitably modeled as a white Gaussian noise process with constant mean. Thus, the E-DSRIF was encoded in Fortran '77 using a second order polynomial dynamical model.

Figures 31 through 35 showed the rt and optical tracker (ot) measurements as predicted by the E-DSRIF. Comparison with the actual measurements in Figures 16, 18, 19, 20 and 22 showed an exact match to within a plotting line width. A better means of comparison is thus provided below in Tables 2 and 3 where r_k^i is range in feet Γ_k^i is azimuth in degrees, and θ_k^i is elevation in degrees. Also, {m,e} corresponds to {actual measurements , estimated measurements}.

The large values of Rk(j,j) for ot variables serves to weight the rt data much more heavily in computing estimates. Decreasing the ot measurement errors to more realistic values should give similar results since the predicted ot measurements matches their actual values very closely.

k	type	r_k^{12}	Γ_k^{12}	θ_k^{12}	r_k^{14}	Γ_k^{14}	θ_k^{14}
23	m	4,355.61	359.18	18.07	4,305.91	357.65	18.04
	e	4,355.61	359.19	18.74	4,355.61	359.19	18.74
24	m	4.683.17	358.91	18.07	4,665.94	358.30	18.09
	e	4,658.07	358.16	18.07	4,658.07	358.16	18.07
.							
.							
.							
229	m	52,171.33	358.96	11.88	52,163.32	358.95	11.89
	e	52,167.32	358.96	11.89	52,167.32	358.96	11.89
230	m	52,332.09	358.96	11.84	52,324.25	358.95	11.86
	e	52,328.15	358.96	11.85	52,328.15	358.96	11.85
.							
.							
.							
459	m	80,328.90	359.22	2.96	80,178.71	359.22	3.00
	e	80,254.10	359.22	2.98	80,254.10	359.22	2.98

Table 2: Radar #350 and #394 Measurements and Estimated Measurements for MLRS. $R_k(j,j)=10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively.
$Q_k(j,j)$ = diag [1.7×10^{17} 7.4×10^{17} 9.3×10^{17}] .

k	type	r_k^3	θ_k^3	r_k^4	θ_k^4	r_k^1	θ_k^1
23	m	154.96	1.05	151.09	3.29	257.38	3.80
	e	154.97	.18	151.87	2.01	255.49	2.88
24	m	154.39	1.19	150.73	3.58	258.16	4.07
	e	154.90	.25	151.71	2.14	256.30	3.00
.							
.							
.							
229	m	68.57	28.54	22.78	21.75	***	***
	e	69.20	26.13	22.93	20.28	331.64	12.52
230	m	***	***	22.64	21.65	***	***
	e	68.74	26.07	22.80	20.18	331.73	12.48
.							
.							
.							
459	m	26.07	5.58	***	***	***	***
	e	26.32	3.94	10.98	3.66	342.28	2.98

Table 3: G30, G80, G110 Measurements and Estimated Measurements for MLRS. $R_k(j,j)=10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively. $Q_k(j,j) = diag\ [\ 1.7 \times 10^{17} \quad 7.4 \times 10^{17} \quad 9.3 \times 10^{17}\ ]$.

Figures 36 and 37 showed the global position estimates and corresponding estimate error covariances respectively. Again, the rocket positions derived from radar #350 as compared with the E-DSRIF estimates based upon all of the 5 selected sensors, showed extremely close agreement. The slight difference in the estimate of height is due to using $T^{launch} = I$ instead of its correct value as defined by equation (34) from [2]. In Table 4 below, the estimates are compared using the correct coordinate transformation.

k	tp	x_k^1	x_k^2	x_k^3	x_k^4	x_k^5	x_k^6	x_k^7	x_k^8	x_k^9
22	e	-14,104.	-243,740.	3,754.	-296.	3,110.	1,015.	430.	-23.	73.
	m	..... same								
23	e	-14,189.	-243,437.	3,798.	-280.	3,104.	995.	429.	-23.	73.
	m	-14,076.	-243,413.	3,806.	-254.	3,120.	986.	436.	-23.	80.
24	e	-14,184.	-243,110.	3,903.	106.	3,271.	1,062.	497.	11.	85.
	m	-14,160.	-243,101.	3,905.	-211.	3,118.	994.	421.	-35.	49.
.										
.										
.										
229	e	-15,018.	-196,406.	12,646.	-19.	1,639.	1.	-.4	-45.	-15.
	m	-15,012.	-196,397.	12,641.	-17.	1,639.	-3.	1.6	-46.	-10.
230	e	-15,020.	-196,242.	12,646.	-20.	1,635.	-3.	-5.	-41.	-28.
	m	-15,014.	-196,233.	12,641.	-18.	1,635.	-4.	1.7	-49.	8.
.										
.										
.										
459	e	-15,197.	-167,388.	5,729.	22.	758.	-431.	21.	-221.	184.
	m	-15,202.	-167,307.	5,704.	8.	866.	-439.	.9	-25.	59.

Table 4: Global Position Estimates and Derived Measurements for MLRS. $R_k(j,j) = 10^{10}$ for ot variables and 10.,1.,1. for rt range, azimuth and elevation variables respectively. $Q_k(j,j) =$ diag [1.7×10^{17} 7.4×10^{17} 9.3×10^{17}] .

Finally, the monotonically increasing estimate error covariance (actually, $P_0 =$ diag [10^{12} ft^2 ... 10^{12} ft^2/sec^2 ... 10^{13} ft^2/sec^4 ...] was used to initialize the covariance propagation so that the first step is a large, off scale decrease to approximately 100 ft^2) is due to our using values of Q_k approximately 12 orders of magnitude higher than its precomputed sample value. A more realistic value should result in a P_k with quite the opposite behavior.

2.2.1 New Results

The E-DSRIF was successfully used to track an MLRS data set in previous work however, a monotonically increasing estimate error covariance was observed (see Figure 37 of [2]). This is due to our using values of Q_k many orders of magnitude higher than its precomputed sample value in order to compensate for any errors in the model. For this subtask, Q_k was adjusted interactively until an acceptable covariance function was obtained. As Figures A through C indicate, more realistic values of Q_k results in a P_k with quite the opposite (and more desirable) behavior previously illustrated.

Q_k starts high at 10^3 in Figures A through C and decreases monotonically to 10^{-18}. Both the position and velocity estimate error variances are still unstable when $Q_k = 10^3$ however both reach stability when $Q_k = 10^{-2}$. Not much improvement is gained by decreasing Q_k beyond 10^{-3}. One of the measurement variables from each of the 5 sensors selected were also plotted (not shown here) in Figures A through C. This was done in order to see whether the oscillations in estimate error variances, which begin to occur when $Q_k = 10^{-2}$, are correlated with the time intervals of data drop-out.

Next, the ot measurement errors were kept artificially high in Phase I. These large values of R_k for ot variables serves to weight the rt data much more heavily in computing estimates. In Figures D through F (where $Q_k=10^1$) and G through I (where $Q_k = 10^{-3}$), R_k was decreased for 3 of the 6 ot measurement variables. Decreasing some of the ot measurement errors from 10^{10} to more realistic values near 1 improved the results significantly by increasing the rate of convergence, especially near the initial time point.

Finally, an adaptive form of the E-DSRIF wherein

$$Q_k = \text{factor} * P_k(+) \tag{97}$$

was encoded. This simple method for tuning the filter by using a feedback loop to compute Q_k, gave exceedingly good results as shown in Figures J through L (where all measurement errors were chosen to be the nominal Phase I values). Best results were obtained using a scale factor of .001, and this was used in the last set of Figures M through O. In this latter set, again R_k was manually decreased for 3 of the 6 ot measurement variables. Overall, the best results (smallest estimate errors) are shown in Figures M and O, for the case $R_k(j,j) = 10^{-1}$ for ot variables. Adaptation by direct feedback of $P_k(+)$ is a less sophisticated means of estimating Qk in real time than the method of maximum likelihood albeit, it is much simpler to implement.

3.0 Estimates of Technical Feasibility

VLSI (Very Large Scale Integration) technology has been developed to the point where high speed floating point processors may be concatenated to form

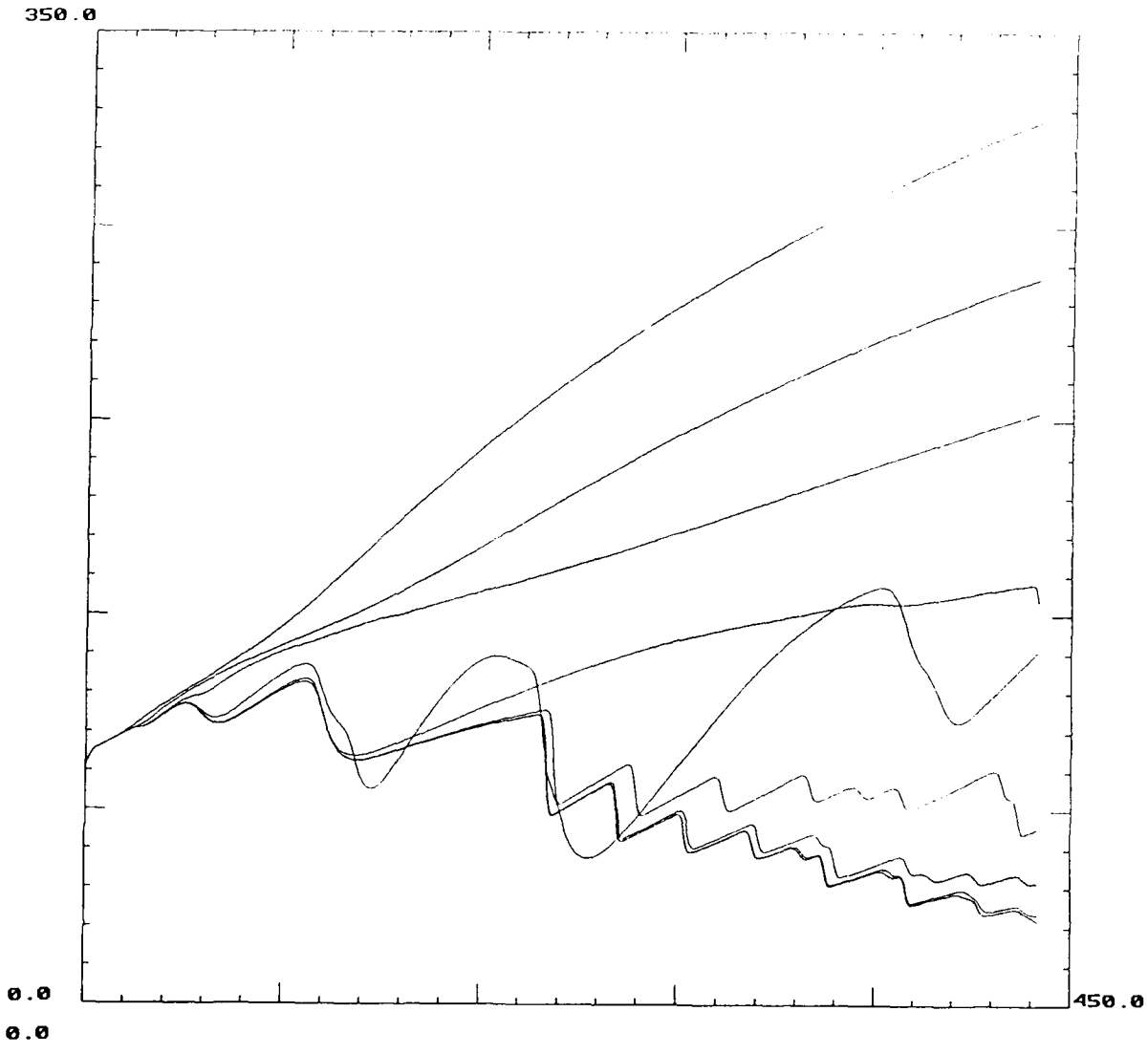


Figure A

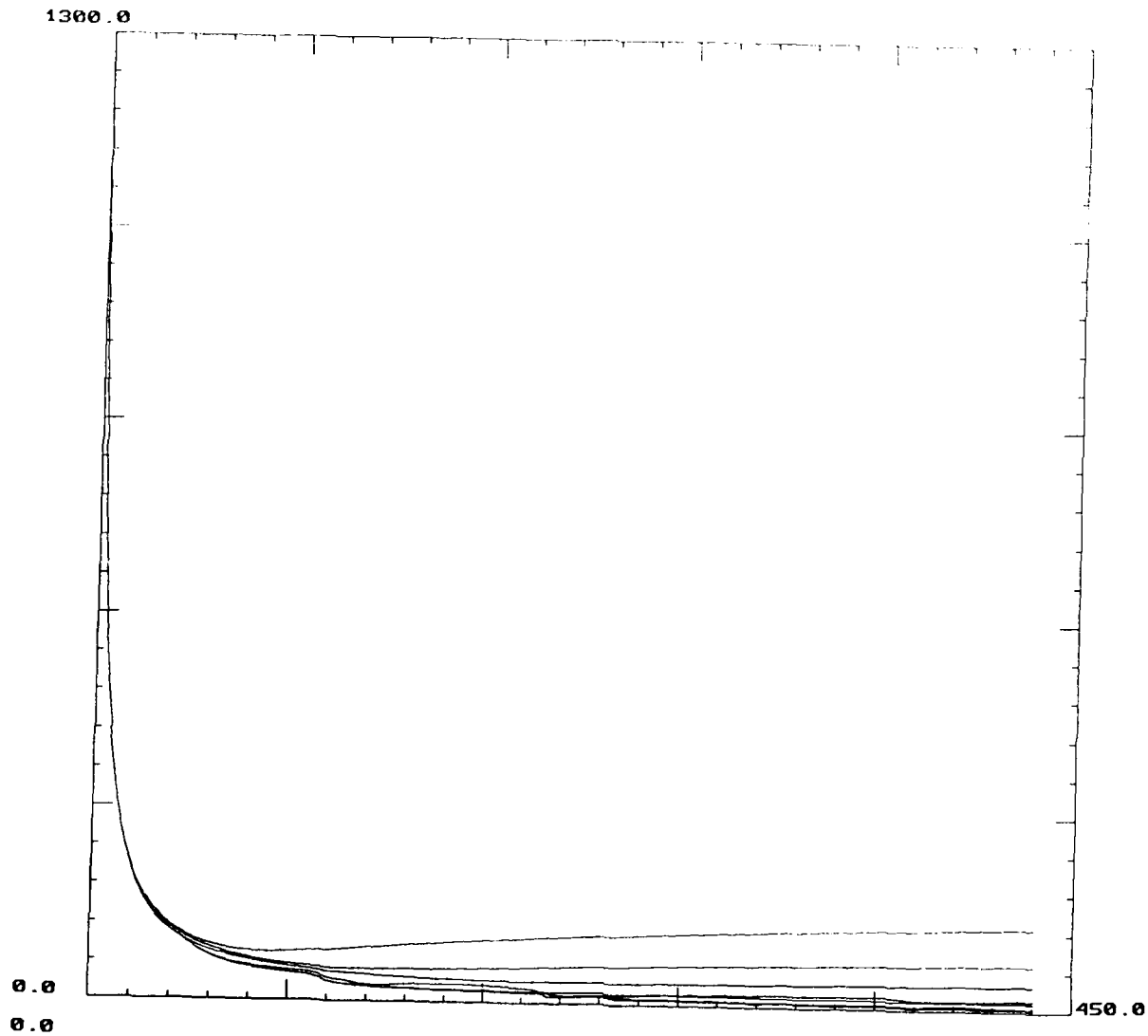


Figure B

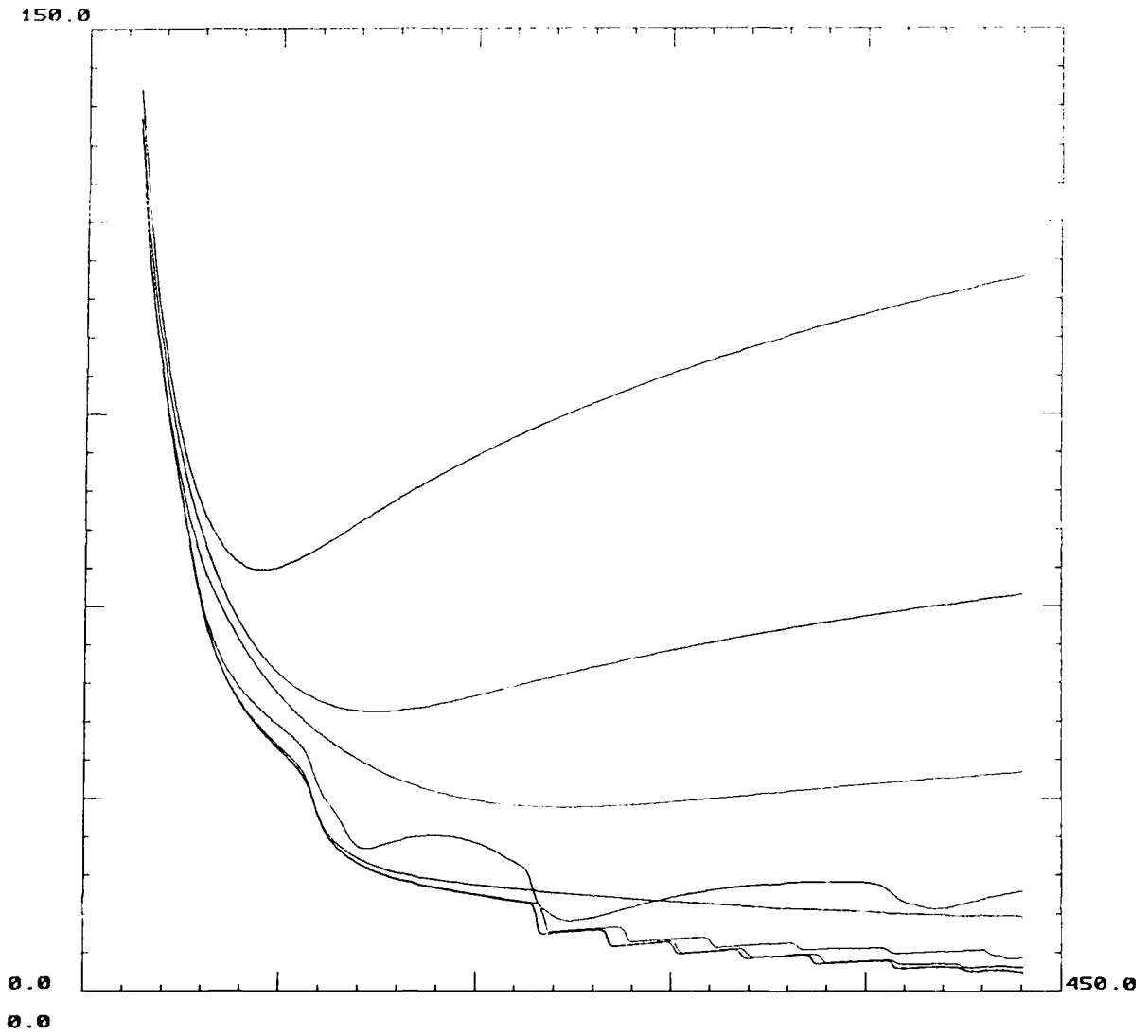


Figure C

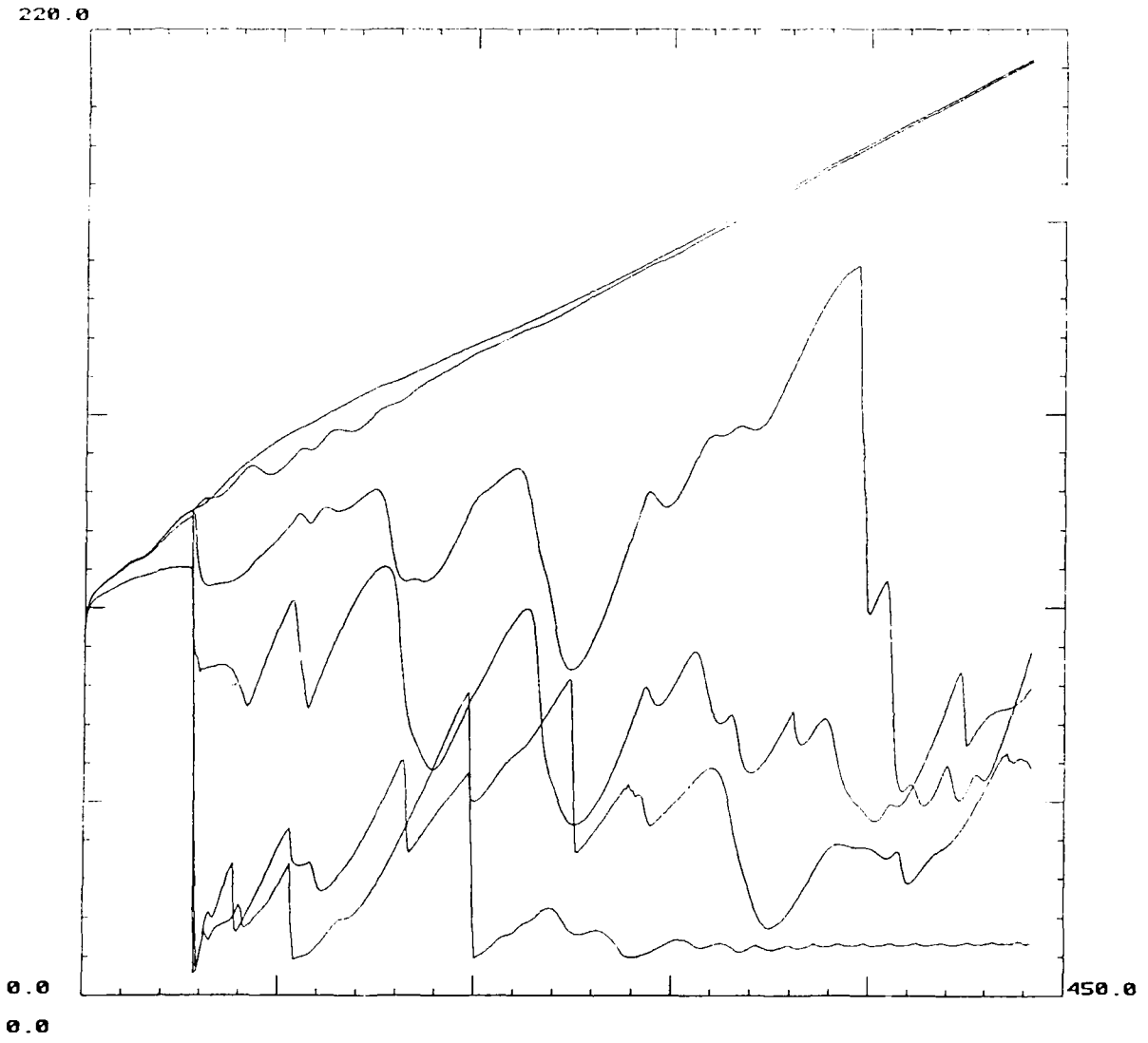


Figure D

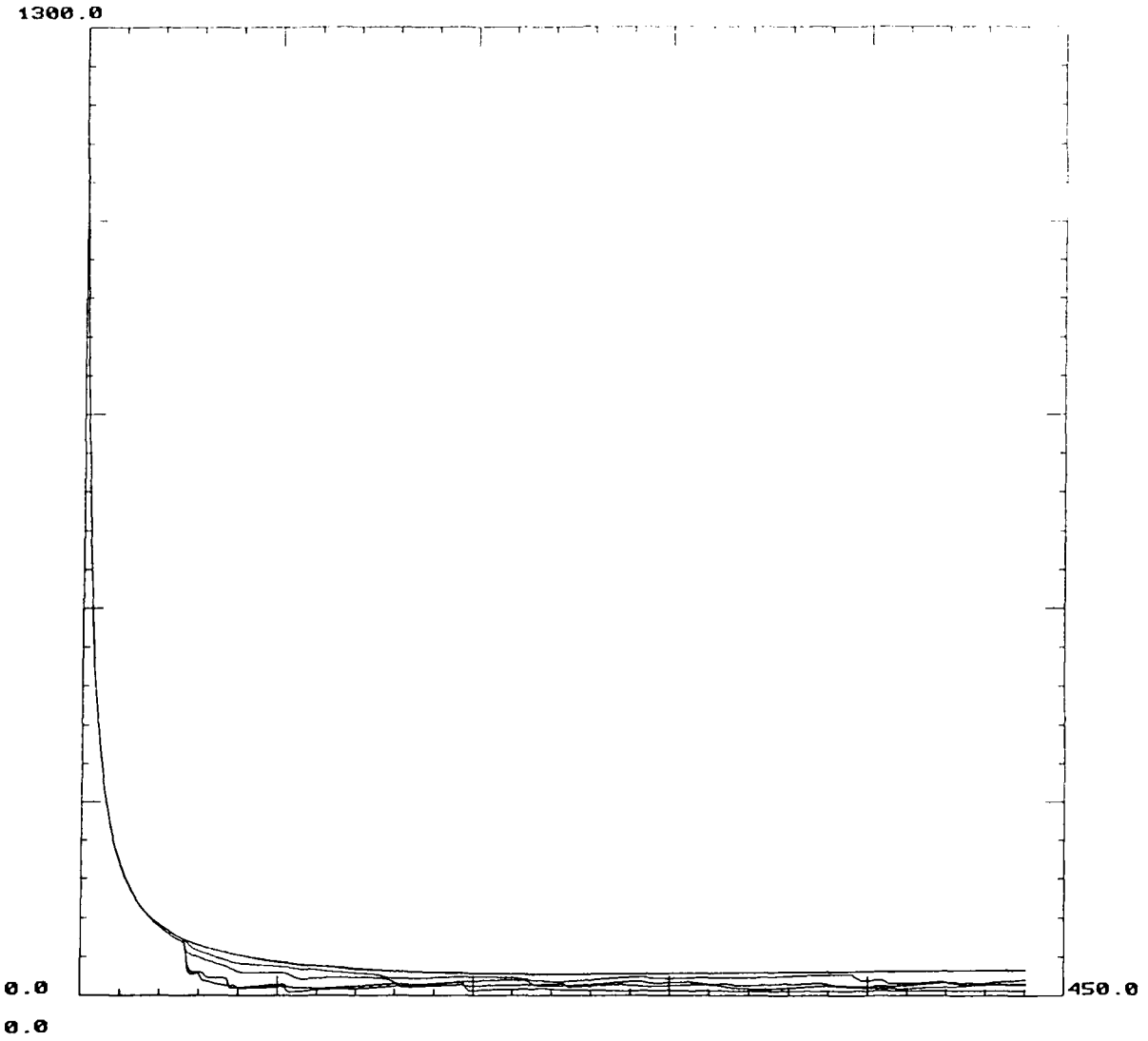


Figure E

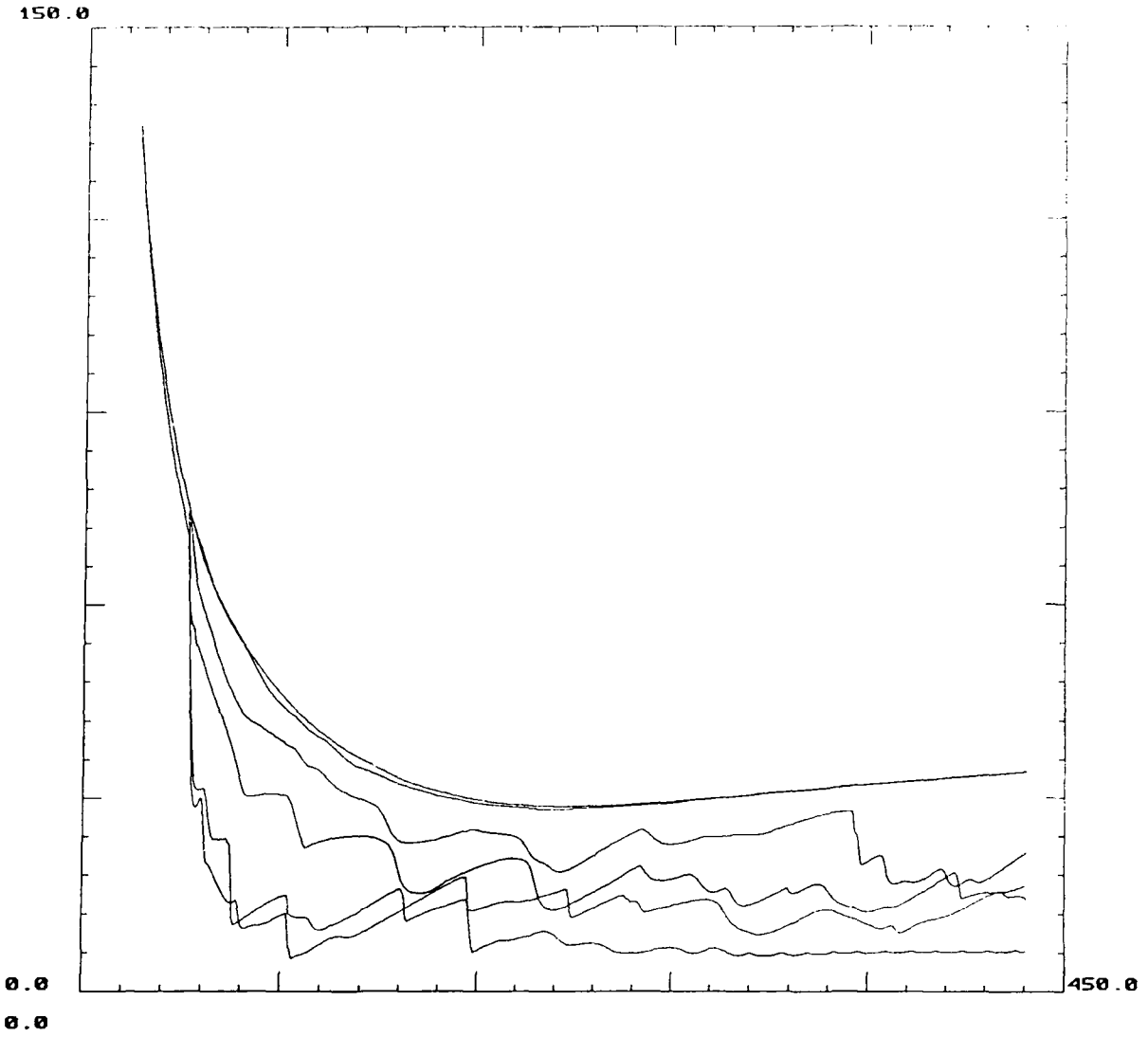


Figure F

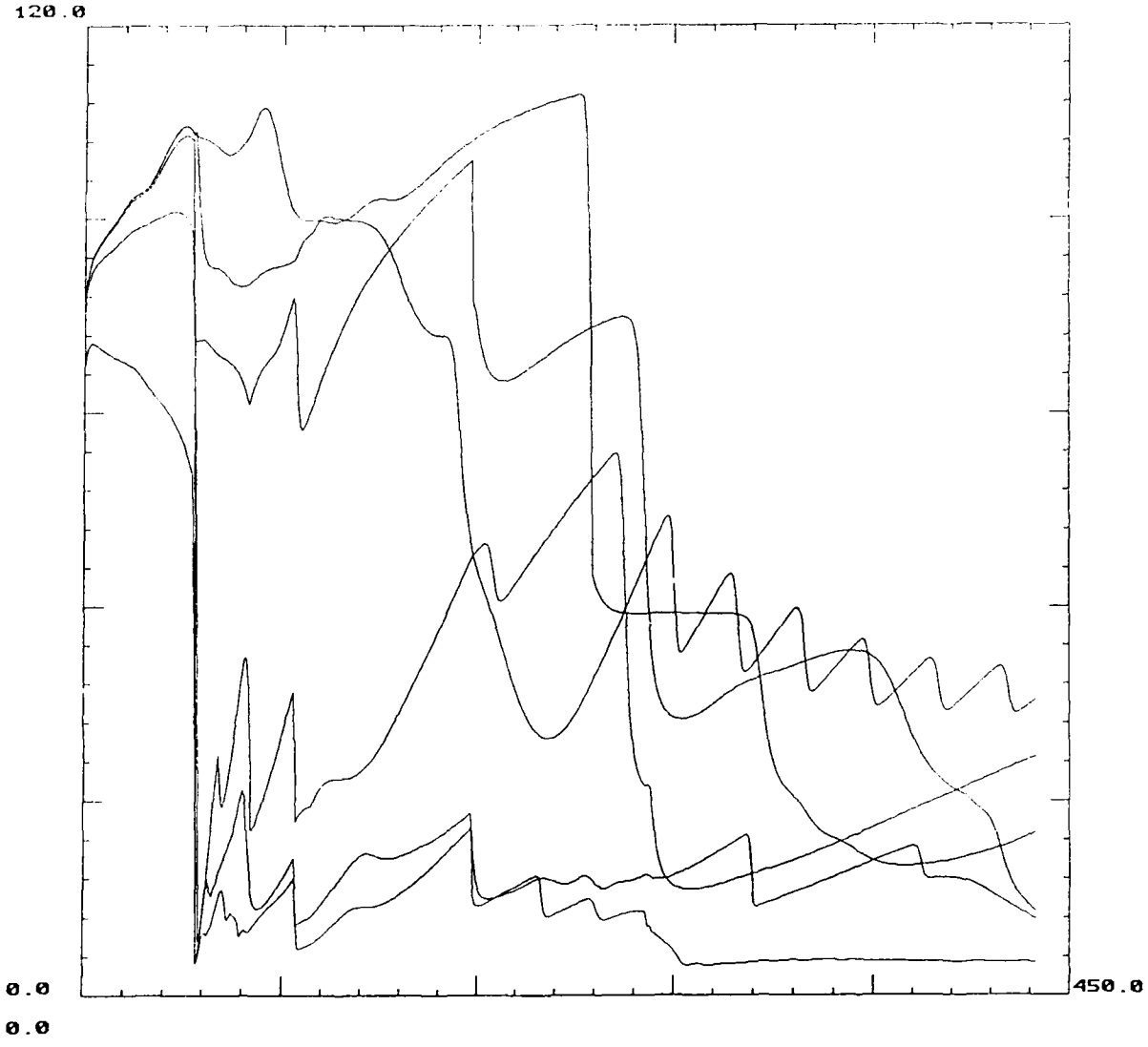


Figure G

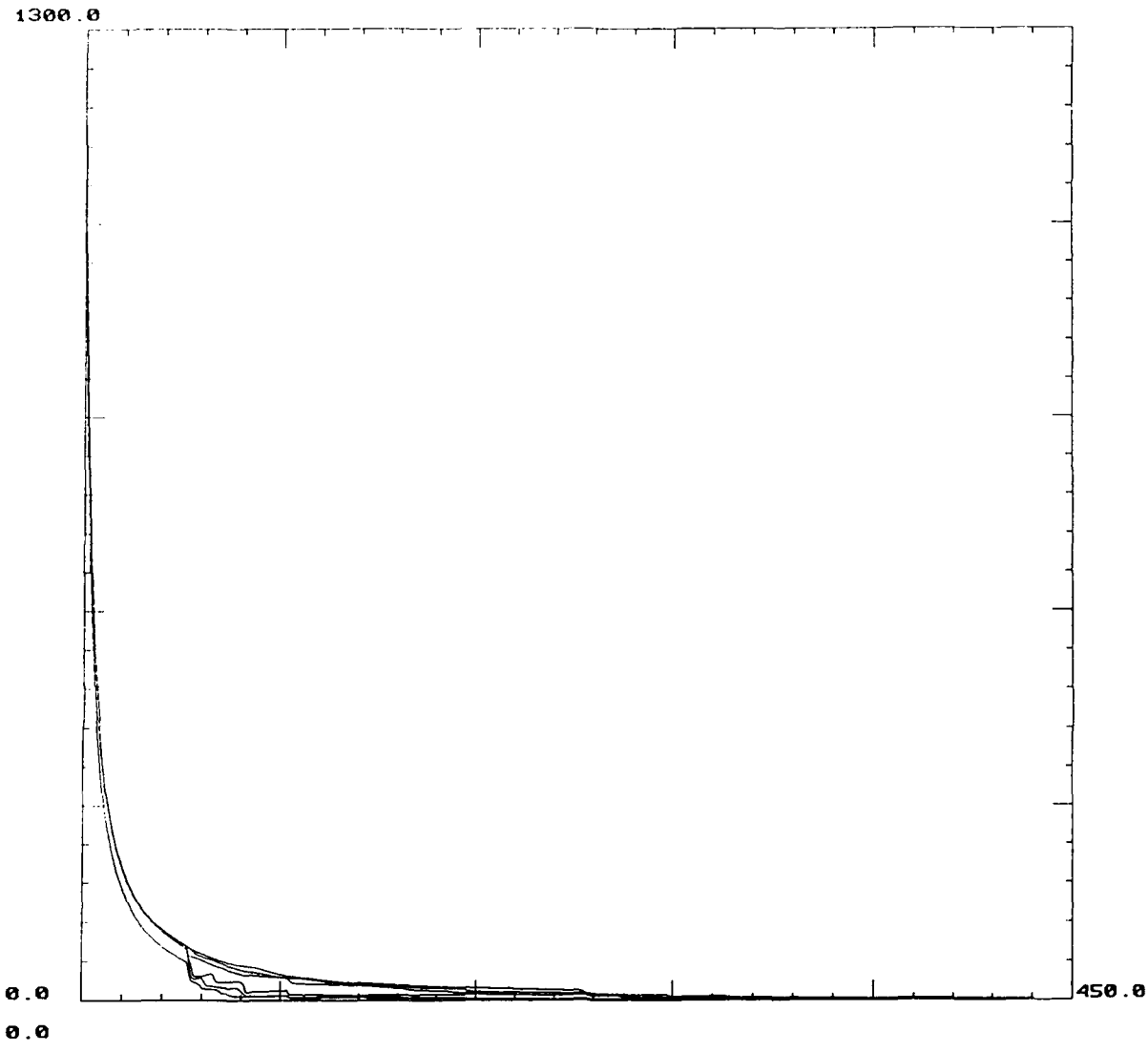


Figure H

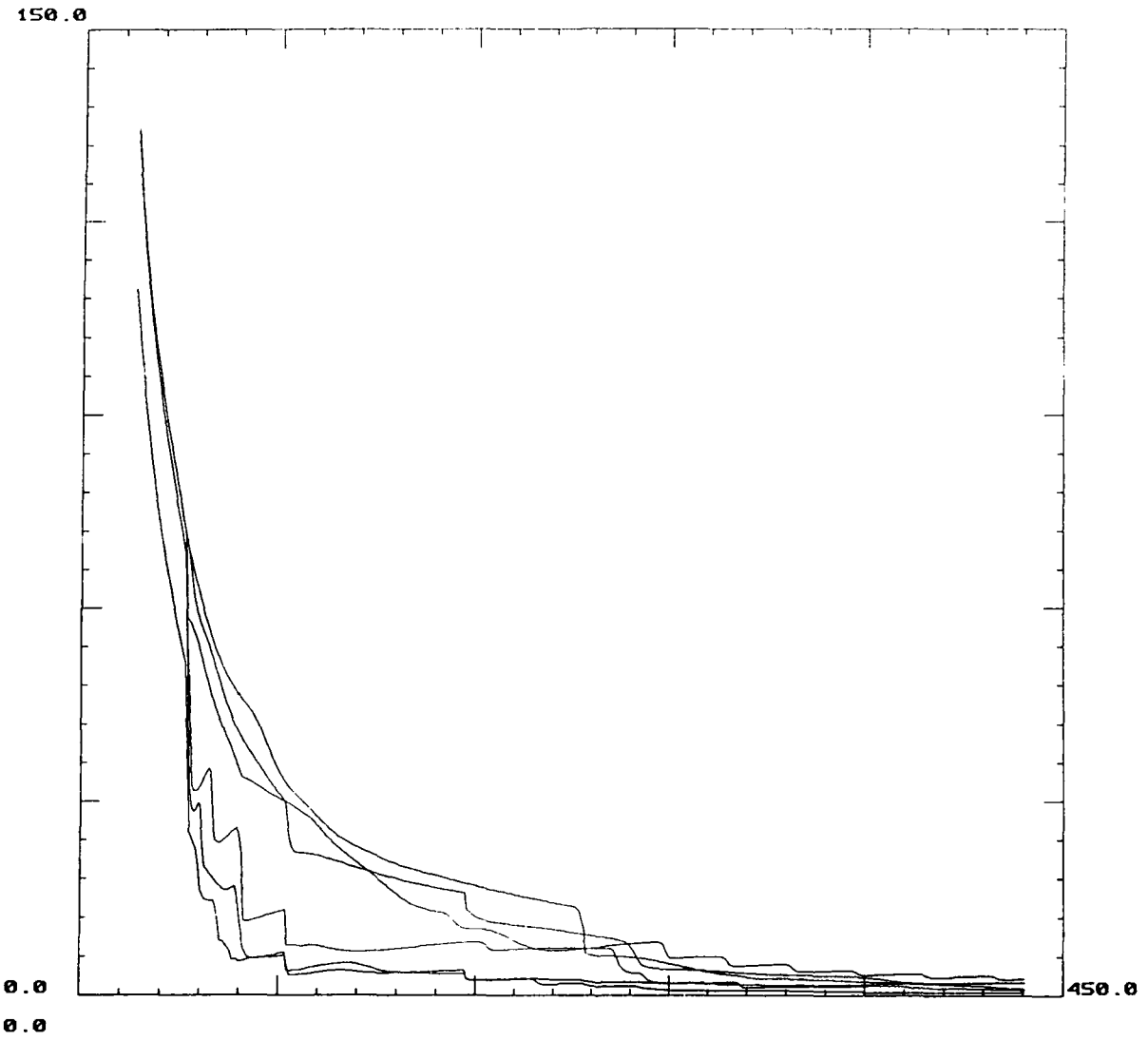


Figure I

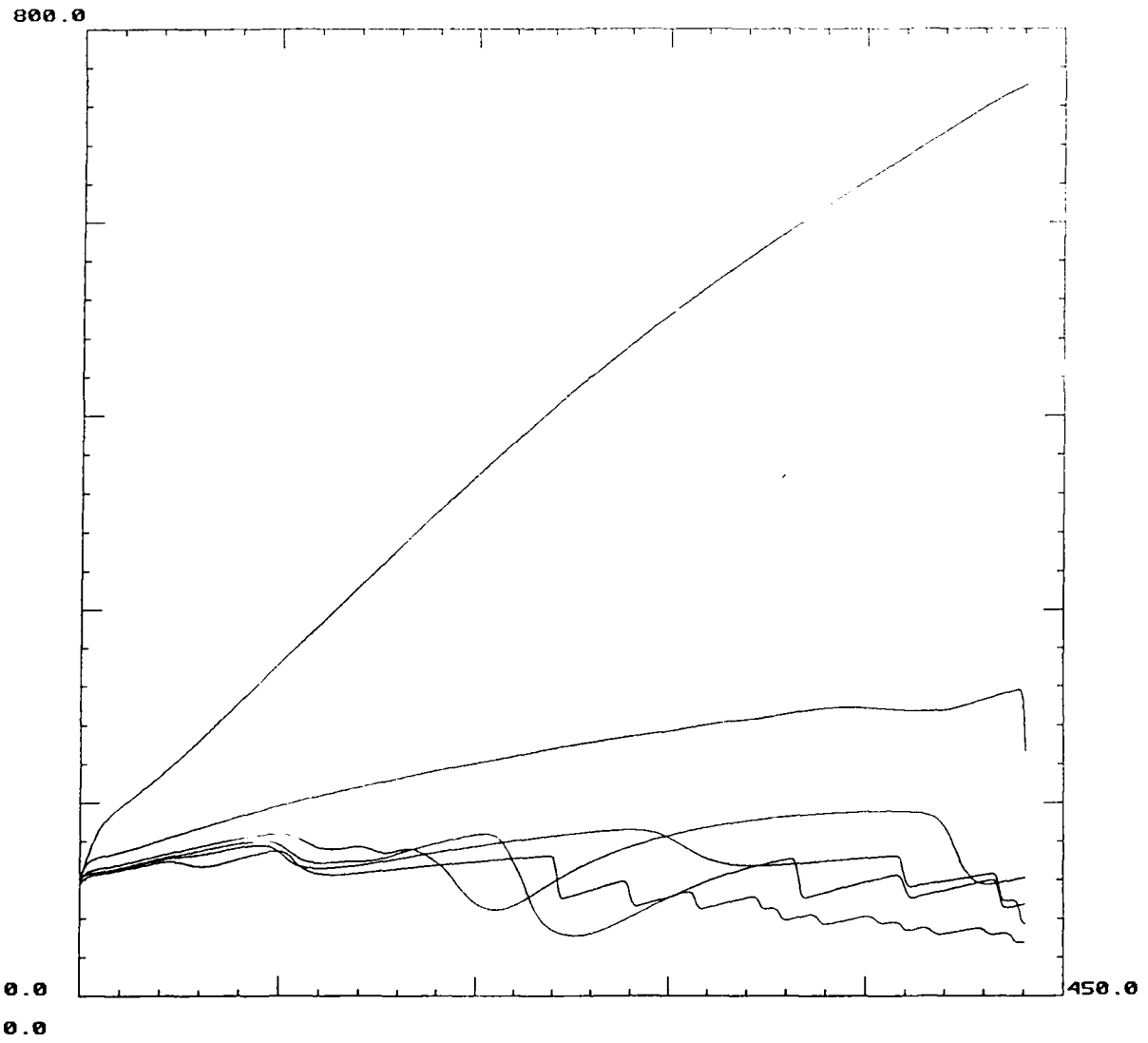


Figure J

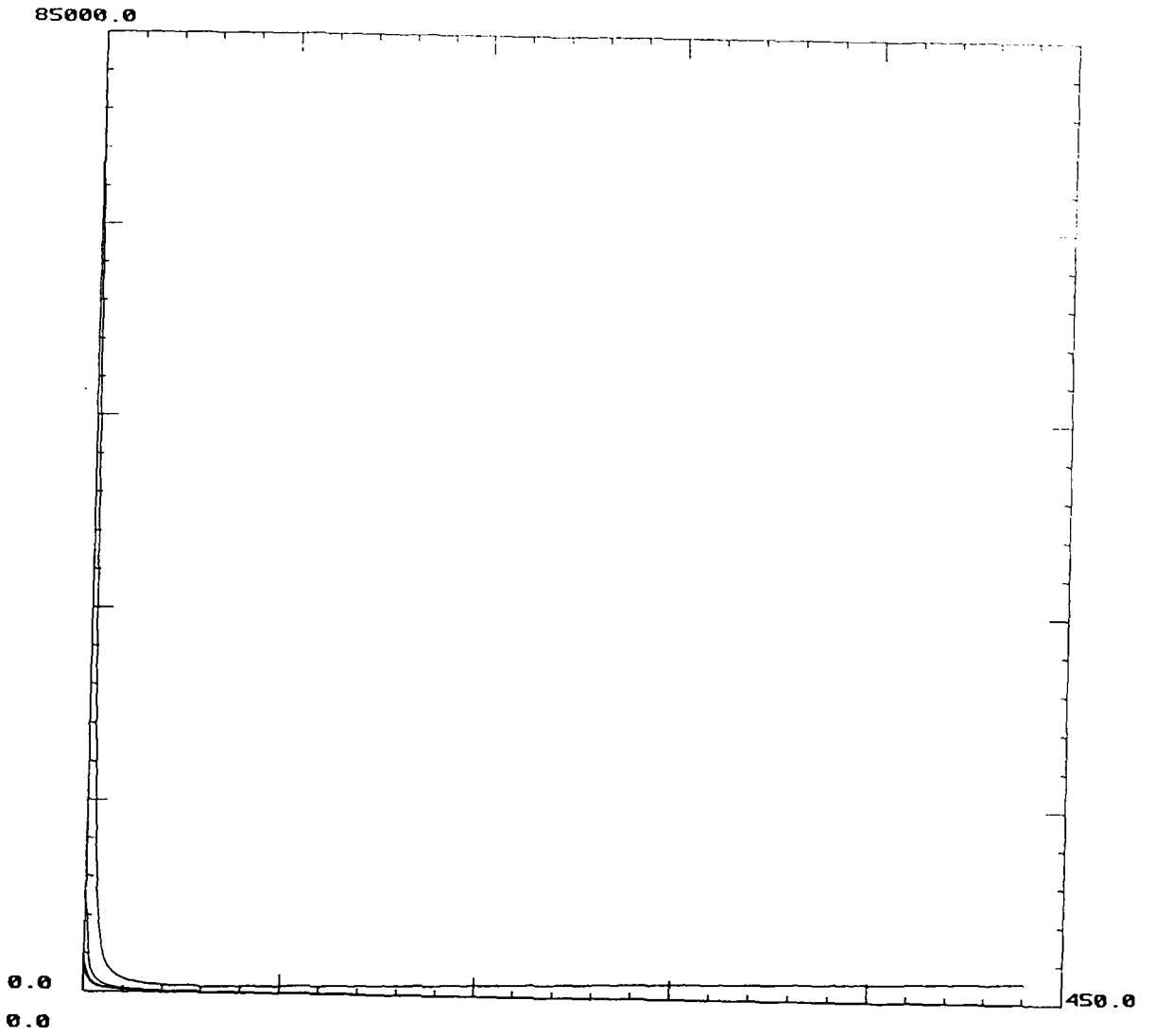


Figure K

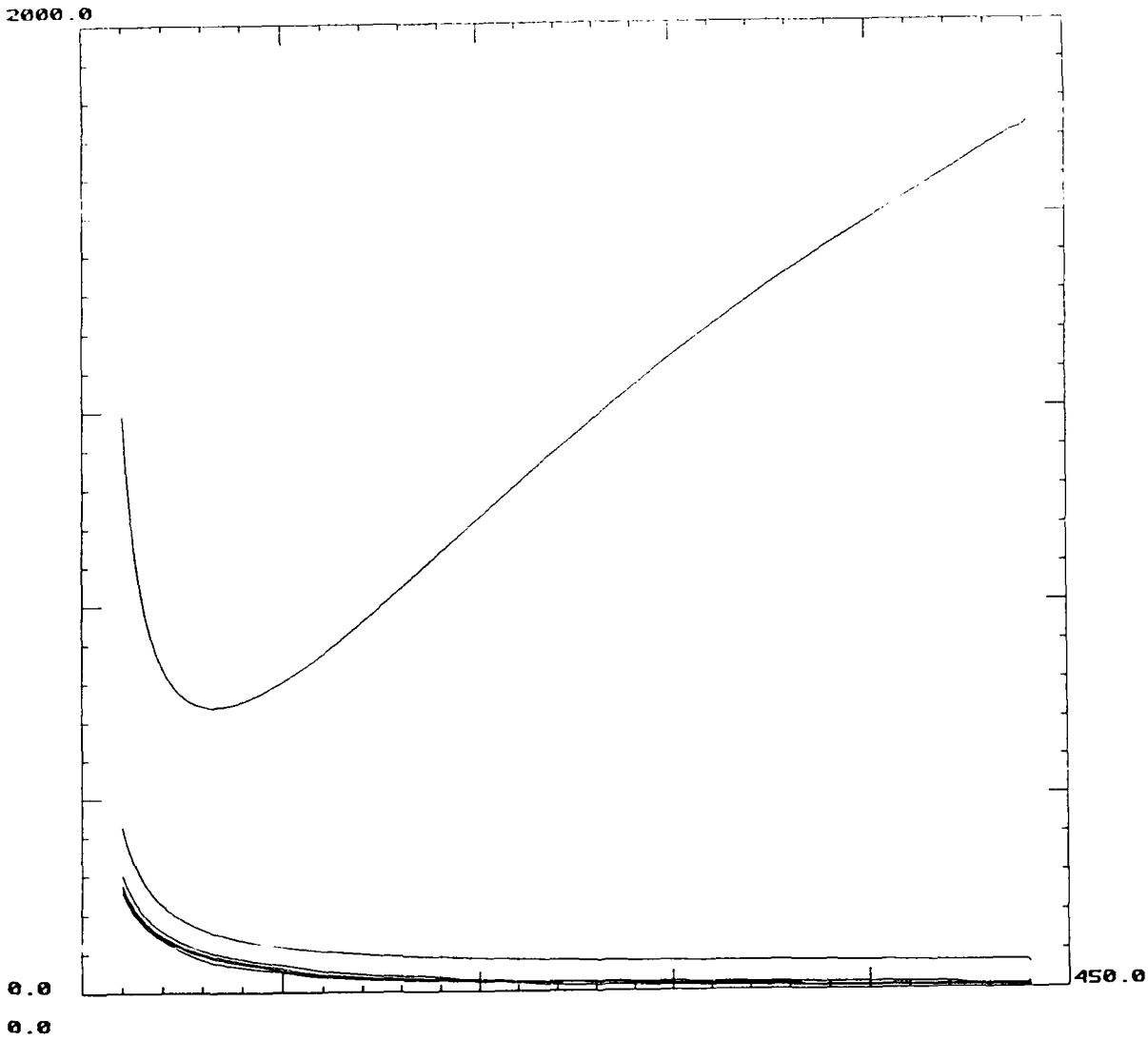


Figure L

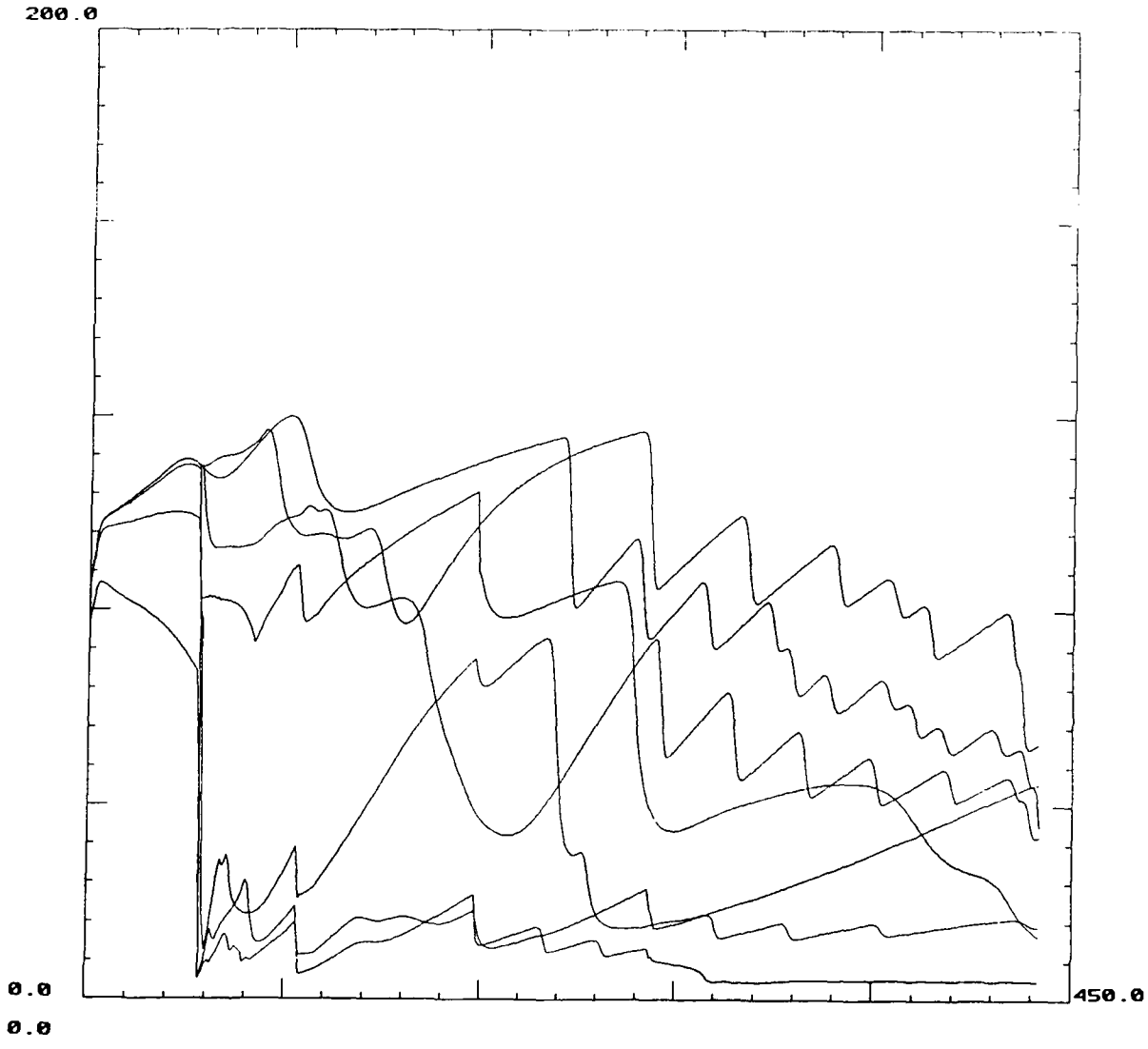


Figure M

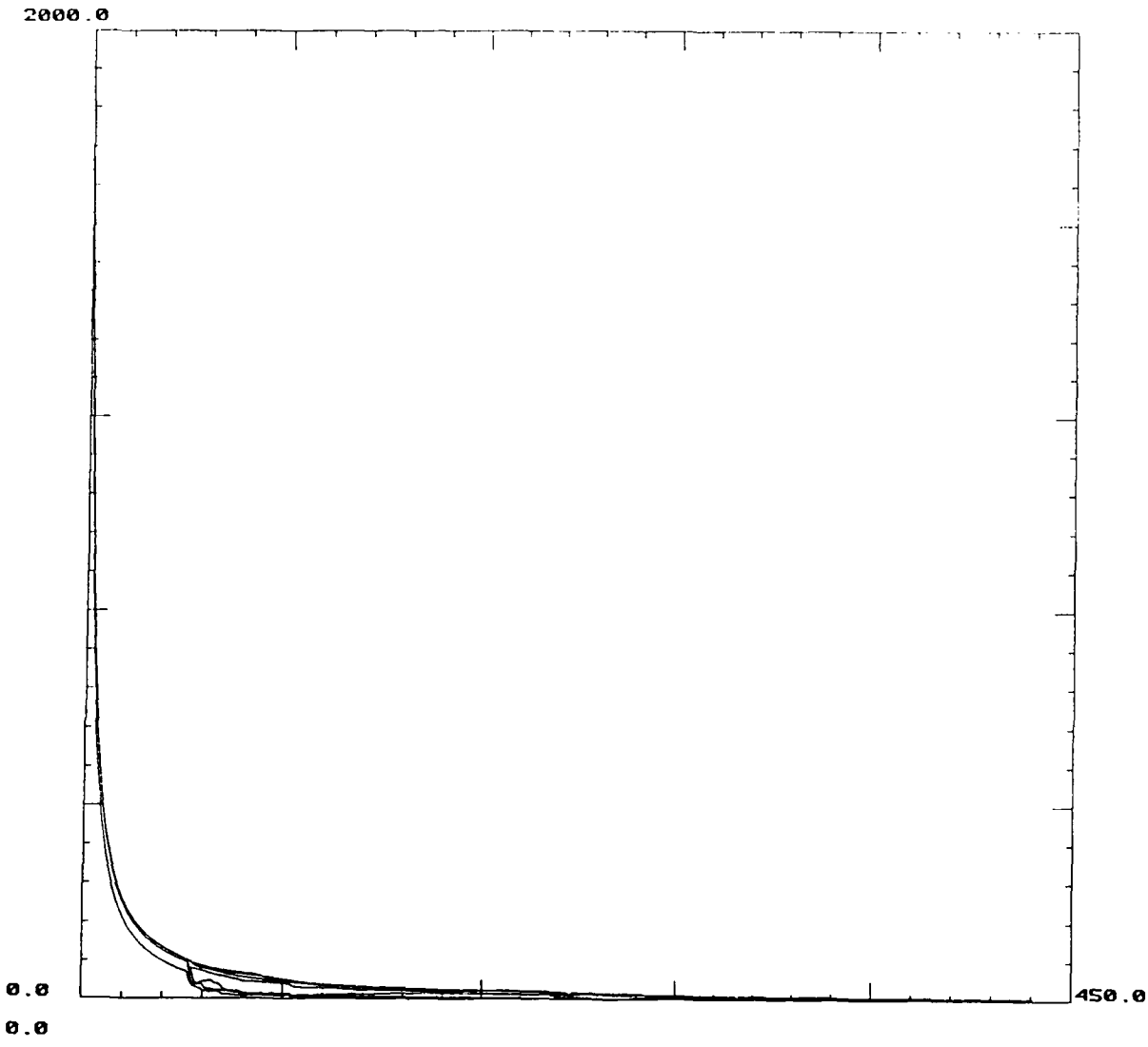


Figure N

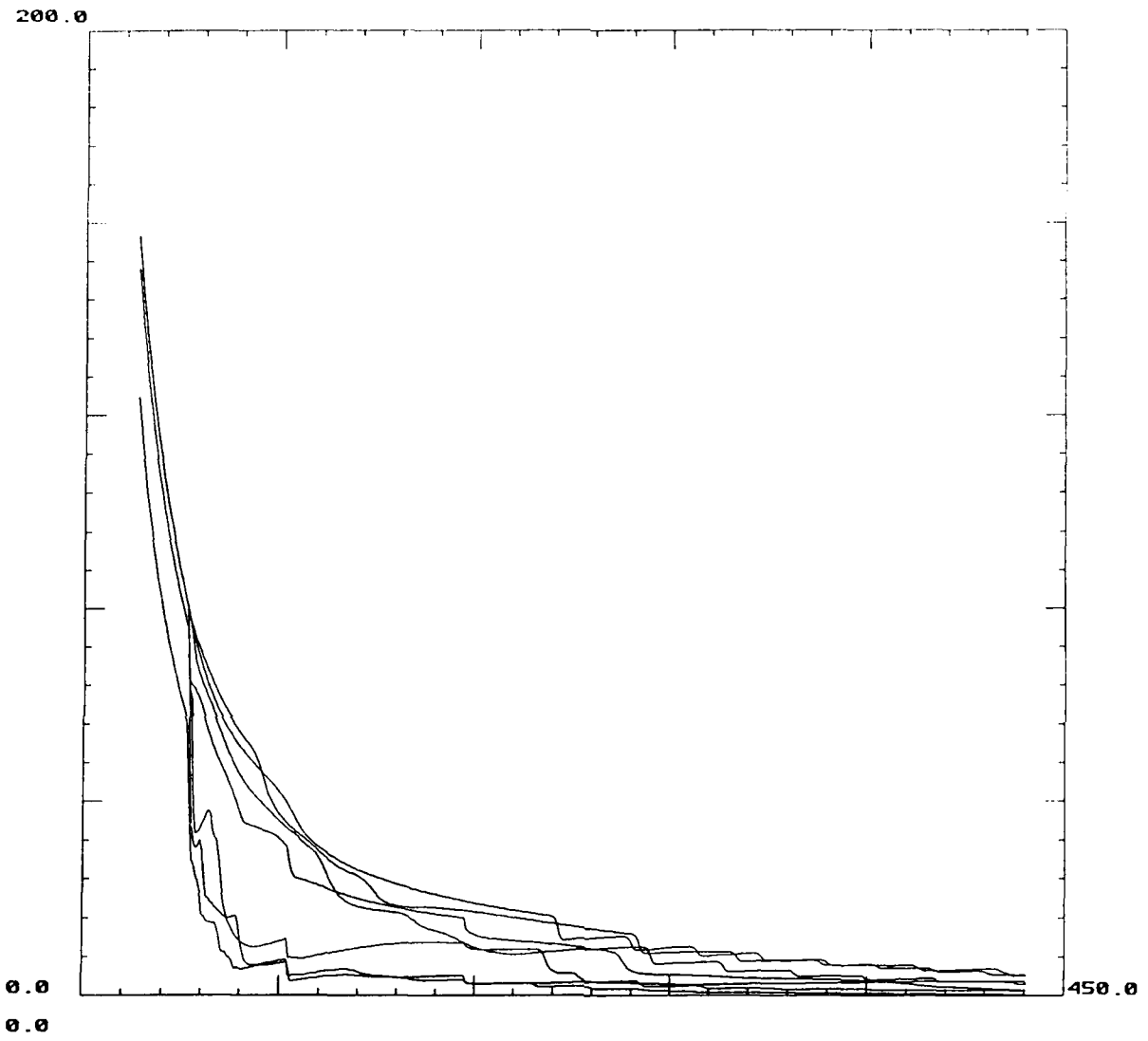


Figure 0

compact supercomputers with far greater throughput than uniprocessor machines. Thus, there is considerable interest within the signal processing community in the development of parallel versions of conventional algorithms. MTI has collaborated with Dr. G.J. Bierman to develop a parallel form of the Kalman filter that has several very unique and important features. We believe that utilization of these features will result in the design of an integrated tracking system that exhibits much improved performance over any "isolated" approach to undersea surveillance or test range tracking.

Specifically, our Decentralized Square Root Information Filter (DSRIF) [1] allows each group of measurement variables, the process noise statistics and the prior information about the initial state to be processed in separate but locally optimal filters. Globally optimal state estimates and estimate error covariances may then be computed by combining local filter outputs on demand. This will allow the analyst to identify the contribution of each measurement group, the process noise and the prior information about the initial state to the global state estimate and estimate error covariance without additional computation.

Furthermore, the process noise and prior information may be distributed amongst the data processing filters in order to improve upon the fault tolerant characteristics of the nominal algorithm when real-time signal processing is an issue. In this case, the estimates and covariances should gracefully degrade from global optimality as local processors fail. Thirdly, the algorithm is based upon numerically reliable matrix factorization methods which, unlike the CKF, will never fail.

The objective of Phase I research was to validate the DSRIF equations by testing its ability to track both predetermined and unknown trajectories when perturbed by white Gaussian noise. The state estimates and error covariances obtained were found to be identical (when printed to 10 significant digits) with those of a SRIF implemented in centralized form with all calculations performed in double precision arithmetic. Furthermore, an adaptive version of the extended DSRIF (E-DSRIF) was successfully used to track real Multiple Rocket Launch System data obtained from the WSMR.

In order to determine the feasibility of our distributed approach to multisensor tracking, several specific technical objectives must be met. First and foremost, the basic DSRIF theory needs to be extended to enable the tracking of multiple targets. This requires that a theory for associating data with targets, based upon the DSRIF (and not the SRIF which already exists, see [2]), be developed. Correlation of measurements with targets can best be done using a hypothesis testing approach. The idea is to select the correlation of measurements with targets that has maximum probability given the data. Calculation of all combinations to form the entire set of these conditional probabilities can be prohibiting, especially in a dense target environment. A major advantage in using the DSRIF is the tremendous reduction in computational cost associated with this calculation.

Secondly, the DSRIF is a new algorithm which has undergone only limited testing in Phase I research. Extensive testing within a multisensor multitarget tracking scenario is needed. Other theoretical questions such as the development of a delayed-state DSRIF for processing range-rate measurements, a method for isolating faulty sensors, and efficient

implementations of the DSRIF that facilitate high data rates need to be addressed.

Finally, consideration needs to be given to the design of the tracking network both at the global and local levels. The major question here is whether a sufficient data rate can be achieved using current chip technology. Another question is whether the architecture can be reconfigured (in software) to implement other members of the family of DSRIFs. A multitude of test range scenarios is envisioned so that a robust system is needed. At one extreme, test vehicles may include slow moving submarines with well defined nominal trajectories a priori while at the other, multiple smart torpedoes with maneuvering capability is possible. The key to a successful network design is to employ a more or less sophisticated version of the algorithm depending upon the particular scenario. Thus the network must be adaptable. For example, preflight simulations of the proposed shot using high fidelity hydrodynamic models can yield good values for the process noise levels and a basic DSRIF should result in good tracking performance. However, a sudden departure from the nominal trajectory would require a detection mechanism as part of the algorithm and adjustment of Q_k in real time. A DSRIF based multisensor laboratory tracking experiment should be performed prior to deployment of a test range prototype.

References

[1] Bierman, G.J. and M.R. Belzer, "A Decentralized Square Root Information Filter/Smoother," Proceedings of the 24th IEEE Conference on Decision and Control, pp. 1902-1905, 1985.

[2] Belzer, M.R. and Y.M. Cho, "Microcomputer Network Architectures for Range Instrumentation Applications," MTI Rept. to White Sands Missile Range, New Mexico, 1988.

[3] Athans, M., R. Wishner and A. Bertolini, "Suboptimal State Estimation for Continuous Time Nonlinear Systems from Discrete Noisy Measurements," IEEE Transactions on Automatic Control, Vol. AC-13, 1968.

[4] Wishner, R.P., J.A. Tabaczynski and M. Athans, "A Comparison of Three Nonlinear Filters," Automatica, Vol. 5, 1969.

[5] Alspack, D.L. and H.W. Sorenson, "Nonlinear Bayesian Estimation using Gaussian Sum Approximations," IEEE Transactions on Automatic Control, Vol. AC-17, 1972.

[6] Schwartz, L. and E.B. Stear, "A Computational Comparison of Several Nonlinear Filters," IEEE Transactions on Automatic Control, Vol. AC-13, 1968.

[7] Wilsky, A.S., M.G. Bello, D.A. Castanon, B.C. Levy and G.C. Verghese, "Combining and Updating of Local Estimates and Regional Maps Along Sets of One-Dimensional Tracks," IEEE Transactions on Automatic Control, Vol. AC-27, No. 4, 1982.